
Graduate Theses, Dissertations, and Problem Reports

2003

Permeability and strength of artificially controlled porous media

Suresh Pasumarty
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Pasumarty, Suresh, "Permeability and strength of artificially controlled porous media" (2003). *Graduate Theses, Dissertations, and Problem Reports*. 1355.
<https://researchrepository.wvu.edu/etd/1355>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Permeability and Strength of Artificially Controlled Porous Media

Suresh Pasumarty

**Thesis submitted to The College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of**

**Master of Science
In
Mechanical Engineering**

**Ever J. Barbero Ph.D., Chair
Jacky Prucz Ph.D.
Kenneth Means Ph.D.**

Department of Mechanical and Aerospace Engineering

**Morgantown, West Virginia
2003**

Keywords: Permeability, Stress concentration, porosity, FEM, SDRC- IDEAS

Copyright 2003 Suresh Pasumarty

ABSTRACT

Permeability and Strength of Artificially Controlled Porous Media

Suresh Pasumarty

Oil well industry has been using hydraulic fracturing to open new channels and thus increase well productivity. Materials known as Proppants are injected into the fracture to keep the channels conductive after the applied hydraulic pressure is removed. But the proppants used to date are often crushed under the influence of rock closures, which leads to clogging of the channel.

The main objective of this study was to model a material that would be able to withstand the rock closure stresses and also remain conductive. A three-dimensional finite element model was developed to study the effects of the shape and the arrangement of inclusions (voids) in the material. Spherical and cylindrical inclusions were studied. Three different void arrangements were simulated, namely random, cubic and Fermat. Strength and Permeability analysis were performed separately.

The cubic array arrangement of inclusions exhibits high permeability and low stress concentration. The random array exhibits low permeability and high stress concentrations. Fermat array was a realistic arrangement where the values of permeability and stress were bound by those corresponding to cubic and random array. The high stress concentrations prompted studying of crumbling. Crumbling analysis was performed to study the effect of crumbling of thin slivers.

To Family, God, and Friends

Acknowledgements

First of all I would like thank my parents Mr. P.M.M.K. Avadhani, and Mrs. P. Lakshmi Sundari for supporting me and encouraging me all the way.

I would like to express my gratitude to Dr. Ever J. Barbero, my advisor, for his guidance. His knowledge and expertise was a great help without which this project would never have been completed.

I am very thankful to my committee members Dr. Jacky Prucz and Dr. Kenneth Means for their suggestions and comments.

I would like to thank Halliburton Company for funding this project and providing me with an opportunity to enhance my knowledge.

I would also like to acknowledge my officemates, Mr. Karthik Sikkil, Mr. Kevin Ford, Ms. Shalini Rangarajan, Mr. William Jennings Briers III for their valuable suggestions and constructive comments.

I am very grateful to Dr. Gasser Abdulal for helping me through out the course of this project.

I would also like to thank my brother Mr. Ramesh Pasumarty who was always there to help me and encourage me constantly.

Table of Contents

Title Page	i
Abstract	ii
Dedication	iii
Acknowledgements	iv
Table of Contents	v
List of Tables	viii
List of Figures	x
Chapter One: Introduction and Literature Review	1
1.1 Introduction	1
1.2 Literature Review	2
1.2.1 Hydraulic Fracturing	2
1.2.1.1 Proppant Types	4
1.2.1.2 Fracturing Fluids	5
1.2.2 Acid Fracturing	5
Chapter Two: Microstructure	7
2.1 Introduction	8
2.2 Representative Volume Element	8
	v

2.3 Spherical Voids	10
2.3.1 Random Array	10
2.3.2 Cubic Array	12
2.3.3 Fermat Array	13
2.4 Cylindrical Voids	15
2.4.1 Random Array	15
2.4.2 Cubic Array	16
2.4.3 Fermat Array	17
Chapter Three: Permeability	19
3.1 Introduction	19
3.2 Darcy Permeability	19
3.3 CFD Permeability	24
3.4 Verification	29
3.5 Threshold Porosity for Creating a Path	31
3.6 Conclusion	38
Chapter Four: Stress Concentration	40
4.1 Introduction	40
4.2 Stress concentrations	41

4.3 Crumbling	46
Chapter Five: Conclusions and Recommendations	50
5.1 Conclusions	50
5.2 Recommendations	51
References	53
Appendix I – FORTRAN[®] Programs to Generate Random Arrangement of	55
Spherical Voids and Cylindrical voids	
Appendix II – Instructions for Creating and Solving a New Model in I-DEAStm	64
Appendix III - Instructions for Creating and Solving a New Model in Gambittm	82
and Fluenttm	
Appendix IV – MathCAD[®] Program to Determine the Value of k_a	99

List of Tables

Chapter Three

3.5.1 – Threshold porosity for each void arrangement.	34
3.5.2 – Porosity needed to achieve clear path with spherical voids arranged randomly.	35
3.5.3 - Porosity needed to achieve clear path with spherical voids in Cubic array.	35
3.5.4 - Porosity needed to achieve clear path with spherical voids in Fermat array.	35
3.5.5 - Porosity needed to achieve clear path with cylindrical voids arranged randomly in the radial direction.	36
3.5.6 - Porosity needed to achieve clear path with cylindrical voids arranged randomly in the transverse direction.	36
3.5.7 - Porosity needed to achieve clear path with cylindrical voids in cubic array in the radial direction.	37
3.5.8 - Porosity needed to achieve clear path with cylindrical voids in cubic array in the transverse direction.	37

3.5. - Porosity needed to achieve clear path with cylindrical voids in Fermat array in the radial direction.	37
---	-----------

3.5.10 - Porosity needed to achieve clear path with cylindrical voids in Fermat array in the transverse direction.	37
---	-----------

3.6.1 - k_a values of models with cylindrical voids of porosity 40% having Fermat array.	39
--	-----------

Chapter Four

4.2.1 – Stress concentration factors for cylindrical voids arranged randomly @ 50% porosity.	43
---	-----------

4.2.2 - Stress concentration factors for cylindrical voids arranged in cubic array @ 25% porosity.	43
---	-----------

4.2.3 - Stress concentration factors for cylindrical voids arranged in Fermat arrays @ 30% porosity.	43
---	-----------

4.2.4 – Stress concentration factors for spherical voids arranged randomly @ 45% porosity.	44
---	-----------

4.2.5 - Stress concentration factors for spherical voids arranged in cubic array @ 30% porosity.	44
---	-----------

4.2.6 - Stress concentration factors for spherical voids arranged in Fermat arrays @ 40% porosity.	43
---	-----------

List of Figures

Chapter One

1.2.1 - A channel filled with proppants	3
--	----------

Chapter Two

2.2.1 - Representative Volume Element.	9
---	----------

2.3.1 – RVE with spheres arranged in Random array	11
--	-----------

2.3.2 – RVE with spheres arranged in Cubic array	12
---	-----------

2.3.3 – Fermat array	13
-----------------------------	-----------

2.3.4 - RVE with spheres arranged in Fermat array.	14
---	-----------

2.4.1 - RVE with cylinders arranged in random array.	16
---	-----------

2.4.2 - RVE with cylinders arranged in cubic array.	17
--	-----------

2.4.3 - RVE with cylinders arranged in Fermat array.	18
---	-----------

Chapter Three

3.2.1 - Permeability enhancement k_a/k_m of cubic array of cylinders with with $k_v=100*k_m$.	22
--	-----------

3.2.2 - Apparent permeability enhancement k_a/k_m vs. dimensionless void permeability k_v/k_m.	23
3.3.1 - CFD procedure.	25
3.3.2 - Pressure drop when the inlet velocity is specified.	27
3.3.3 - Pressure drop when the inlet mass flow rate is specified.	28
3.4.1 - Cubic array of sand (spherical) particles to simulate permeability of 20-40 mesh sand.	29
3.4.2 - Velocity contour in 20-40 sand model.	30
Chapter Four	
4.2.1 - Stress concentration factor vs. porosity for spherical inclusions.	41
4.2.2 - Stress concentration factor vs. porosity for cylindrical inclusions, $l/d=2$.	42
4.2.3 - Stress concentration factor vs. porosity for spheres with random packing.	43
4.3.1 - Two-void model used to study crumbling.	48
4.3.2 - Effect of yield on stress concentration factor (SCF).	49

Chapter One

Introduction and Literature Review

1.1 Introduction

Reservoir [1] is defined as an accumulation of hydrocarbons in porous permeable sedimentary rocks. When an oil-bearing rock is drilled, the fluid must flow through the pore spaces to reach the well bore and subsequently to the well surface. The fluid will flow if these pores are interconnected forming a “channel”. Deficiency or lack of such channels is a hindrance in the extraction of petroleum from reservoirs. When a rock has few channels or no channels the fluid flow is restricted, and in some cases impossible.

To overcome this undesirable aspect fracturing [2] was developed in 1948. In fracturing, a rock is held together by the stresses acting on the rock and the strength of the rock itself. When the well bore filled with fluid is pressurized, the fluid enters these pores. When hydraulic pressure applied is more than the internal forces acting on the rock, these pores will expand forming a channel. When the applied pressure is removed, the rock forces become predominant and the channel will close. So, to maintain the channels open, some solid material called “proppants” are introduced into the crack. The proppants used till date are expensive and sometimes are unable to withstand the high stresses that act on them when placed in the channel. The proppants usually used in the petroleum industry are sand, sintered bauxite, and ceramics.

1.2 Literature Review

There are two types of fracturing:

(a) Hydraulic fracturing, and

(b) Acid fracturing.

1.2.1 Hydraulic fracturing

Hydraulic fracturing is performed by pumping small particles along with the fracture fluid. “Proppant” is defined as a mixture of fracture fluid and the small particles called propping agent. In hydraulic fracturing, the fracturing fluids (mixed with proppants) are pumped at high pressures. The objective of hydraulic fracturing is to pack the fracture with proppants to ensure that the channel created by the fracturing fluid is intact even after fracturing process is terminated. In order to achieve this, there are two options: pump the fluid continuously, or keep the channels “propped” by packing these channels with proppants. Since pumping the fracturing fluids continuously at high pressures is difficult as well as expensive, propping the channels is the only plausible alternative.

The early fracture fluids were fluids of low viscosity such as water or some low-viscosity oils. But this resulted in rapid settlement of the proppants. A study was performed by Kern and Perkins [10] to predict the mechanism of proppant transport. According to them, the proppant movement was characterized by the formation of a dune at the entrance of the fracture. The proppant would move as soon as the dune attained an

equilibrium height. This packing mode is applicable when the apparent viscosity of the fracture fluid is low. Low viscosities are now attainable by using cross-linked polymer system, and the settlement of proppants can be controlled to achieve desired proppant distribution. One the most important factors to be considered while selecting a fracturing fluid is the capability of the fluid to transport and hold the proppant particles in suspension till the fracture is closed. Fig 1.1 shows how a channel looks after it is filled with proppant.

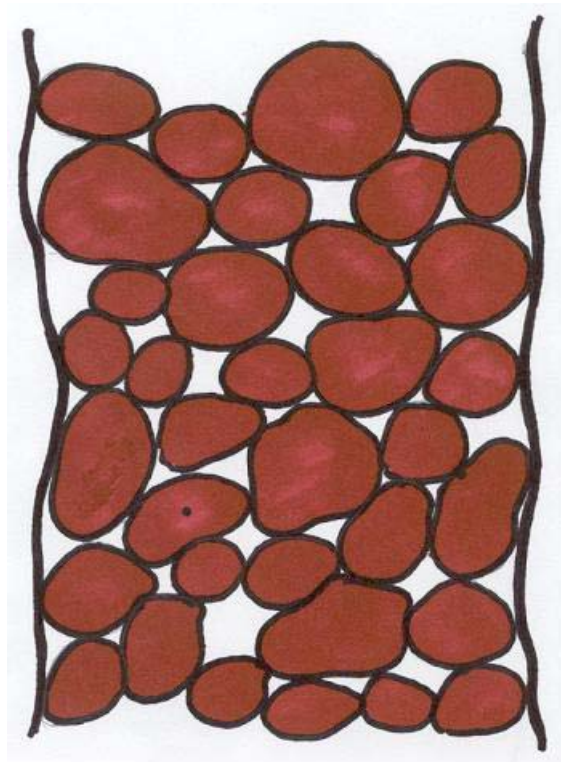


Fig 1.2.1 A channel filled with proppants [6].

1.2.1.1 Proppant Types

Many materials like glass bead, plastic beads and aluminum pellets have been used as proppants since the advent of hydraulic fracturing. The three main proppants that are widely used in the petroleum industry are sand, sintered bauxite, and ceramics.

Sand. Sand is the most primitive, inexpensive and also the most widely used proppant. Since it is not dense like other proppants, the fracture fluids needed to carry sand are of low viscosity. Sand has low compressive strength. When sand is crushed it breaks into smaller fragments rather than turning into powder. This is advantageous because it still maintains high fracture conductivity. But as the rock closure stress increases, the sand becomes more and more compact. This clogs the channels thus effectively decreasing the fracture conductivity.

Sintered Bauxite. The features of sintered bauxite are its high compressive strength (above 1×10^5 kPa) and high density ($\rho_p = 3400\text{-}3800$ kg/m³) [3]. The high compressive strength is a desirable feature since it can resist high rock closure stresses. Because of the high density, fracture fluids of high viscosity are needed to hold the proppant in suspension.

Ceramics. Ceramics are relatively new in the field of proppants. They have higher compressive strength compared to bauxite.

1.2.1.2 Fracturing fluids

Fracturing fluids are important elements of hydraulic fracturing. They carry the proppant and hold them in suspension in the fracture. The typical characteristics of a fracturing fluid are “proper viscosity in fracture, low friction pressure during pumping, provide good-fluid-loss control, break and clean up rapidly once the treatment is over and also be economical”[9].

The early fracturing fluids were oil-based. 1950's saw the advent of water based fluids thickened with gaur. In 1969, the first crosslinked gaur treatment was performed. By 1970, hydroxypropylgaur was being used.

1.2.2 Acid Fracturing

Acid fracturing is defined as “A well stimulation process in which acid, usually hydrochloric acid (HCl), is injected into a carbonate formation at a pressure sufficient to fracture the formation or to open existing natural fractures” [9]. This is a very slow process and has to be performed with great care to prevent water or gas production. The mechanism of this process is simple, acid reacts with the rock, etches it, and forms a permanent channel. Since the acid will not etch uniformly, conductive channels will exist even after the fracture closes.

Acid fracturing is not effective in the treatment of sandstone, because even strong acids cannot etch the fracture surface adequately to produce desired results. So its use is

usually limited to limestone and dolomite surfaces. The fracture length is controlled by the distance the etching acid is able to penetrate before being dead (spent and unable to react).

The two major factors controlling the effectiveness of acid fracturing treatments are the resulting fracture length and conductivity [9]. The resulting fracture length is dependent on acid fluid-loss characteristics, acid penetration distance, and acid flow rate in the fracture (Nierode et al., 1972). Acid flow rate and fracture width are the factors controlling acid spending inside the fracture.

The effectiveness of acid fracturing is also controlled by fracture conductivity. The manner in which the acid reacts with the fracture faces affects fracture conductivity. The acid used in the process does not react uniformly with the fracture surface, so the etching is not uniform. This results in open channels at the end of the treatment.

Chapter Two

Microstructure

Drilling companies have been using proppants to keep conductivity in the channels opened by hydraulic fracturing. The proppants used to this date are not fully effective because they crush under the influence of rock closure forces. So Halliburton Company decided to develop a new material that can be used as proppant. This new material needed to be conductive and also be able to withstand the rock closure stresses. The material was to be analyzed separately for stress concentration and permeability (also referred to as conductivity).

The new material was designed by adding some inclusions to the fluid in slurry state. Next stage is slurry hardness (e.g. as concrete) and finally the inclusions are dissolved, leaving pores in a solid matrix. The best possible arrangement of inclusions and the best shape of each inclusion is determined after performing Computer aided Finite Element Analysis. SDRC® I-DEAS® and Ansys® were used for strength analysis. SDRC® I-DEAS®, Fluent® and Gambit® were used for permeability analysis. Formidable obstacles were encountered during mesh generation, most of which were solved.

The strength of the material was determined by computing stress concentrations. To calculate the permeability two approaches were adopted.

2.1 Introduction

The most important aspect of simulation of artificial porous media is the micro-structure that may be created in practice. There was no experimental precedent as to what micro-structure could be created. The micro-structure had to be realistic but also had to be simple because of limitations of the simulation software. The micro-structure has two main aspects: the void geometry and the packing array. These are described in the following sections.

2.2 Representative Volume Element

It is assumed that the actual structure to be filled with the proposed porous material has dimensions at least an order of magnitude larger than the dimension of the voids. For example, 20-40 mesh sand has a mean diameter of 640 microns. This means, that the smallest gap to be filled has to be at least 6.4 mm. The next assumption is that the geometrical distribution of the voids repeats itself throughout the entire space. If these assumptions are correct, it is not necessary to model the entire structure, but only a representative volume element (RVE). A cubic RVE of dimensions a_i ($i=1...3$) was used (Fig 2.1). The ratio of the smallest a_i to the smallest dimension of the inclusions was about four.

The direction of $(V_i \text{ and } P_i)$, (V_o, P_o) should coincide with the Finite element model.
That is, use X-direction or Y-direction or Z-direction depending on the models (IDEAS and Gambit)

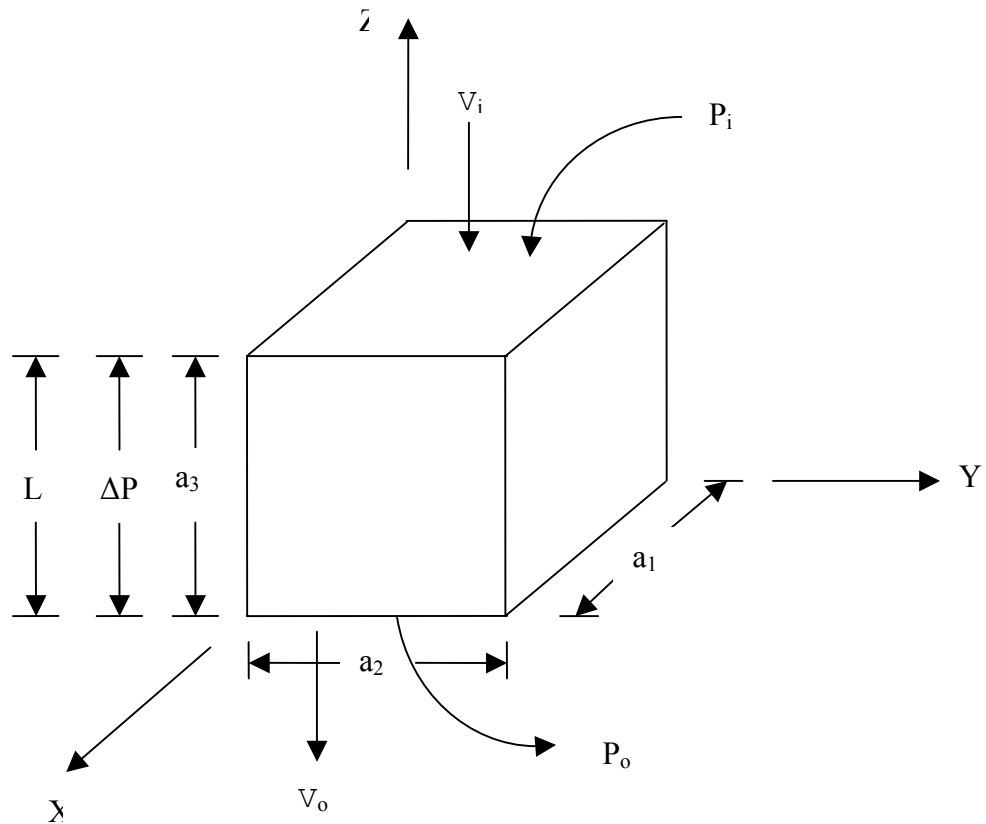


Figure 2.2.1 Representative Volume Element

2.3 *Spherical voids*

The model was initially created with spherical voids because these were believed to be the simplest type of inclusions that could be pumped into the well. The matrix will not be saturated with holes, but they will occupy certain positions inside the RVE. These positions are called packing arrays, although the voids are not packed since they do not saturate the available space.

2.3.1 Random array

Random array is conceptually simple but practically difficult to model. The procedure used for creating truly random placement of spheres was to add, one by one, spheres to the RVE (Fig. 2.2) in positions x_i ($i=1..3$) obtained from a random generator program. The spheres were allowed to overlap a maximum fraction of the sphere diameter. The overlap was denoted by “o” and $o=0.1$ (10% overlap) was used mostly. The overlap “o” can be varied. The overlap was introduced to simulate the effect of surface tension in the matrix when two spherical inclusions touch. Due to surface tension, the liquid matrix will not penetrate completely between the two spheres. Once the inclusions are dissolved, the void space created by two touching spheres would be in the shape of a peanut. The easiest, most effective way of simulating this was to allow for overlap. If a given sphere, as located by the random generator, would overlap more than “o”, the program would reject it, and other random position would be selected. The process is repeated until enough spheres are placed to achieve the target porosity. As the

porosity becomes higher, it becomes more difficult to find positions for the spheres. The parts of the inclusions protruding from the faces of the RVE are removed, thus providing inlets and outlets for fluid flow.

A great shortcoming of random array is that two spheres may end up very close but not touching. This is bad in terms of permeability because no flow would go from one void to the other. It is also bad in terms of stress concentration because the thin layer of matrix between the two voids is very weak and thus lead to high stress concentrations. Furthermore, it is very unlikely in practice because solid inclusions will tend to push each other and touch during mixing.

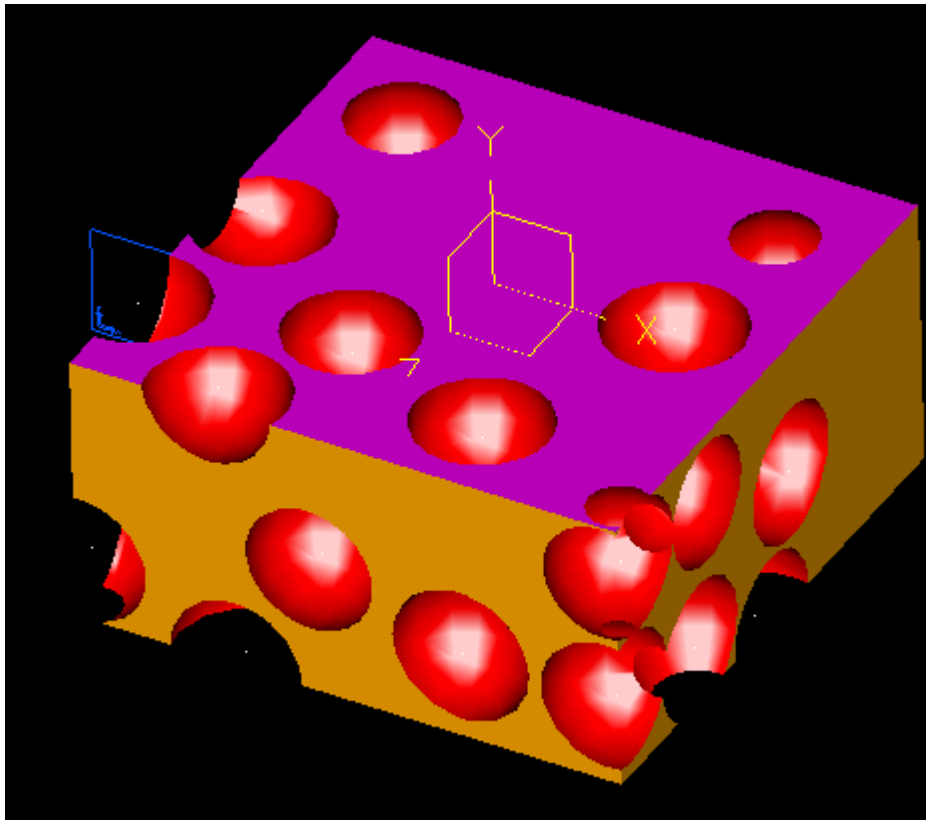


Figure 2.3.1 RVE with spheres arranged in Random array

2.3.2 Cubic array

The cubic array is unlikely in practice but it constitutes an upper bound for ordering. Thus, it was deemed worth studying. The cubic array is generated by starting with a cubic grid of dimensions $d-o$ where “ d ” is the sphere diameter and “ o ” is the overlap. A random generator picks positions in the grid so that voids can be placed randomly in the grid (Fig. 2.3) until the target porosity is reached.

The cubic array has excellent permeability and low stress concentration due to its ordered nature. It helped us prove the importance of “order” to achieve low stress concentrations and high permeability.

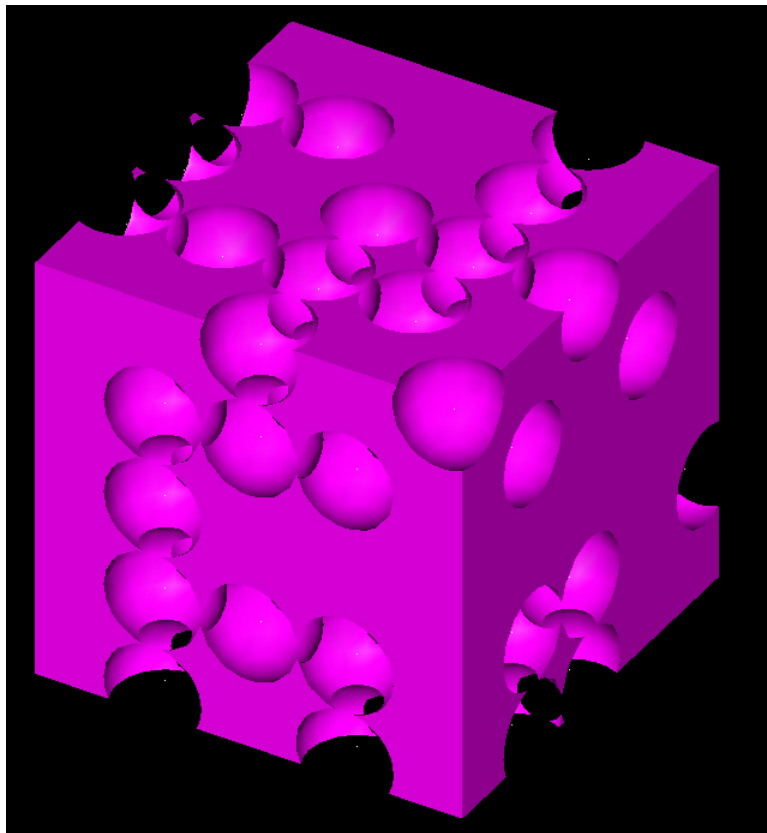


Figure 2.3.2. RVE with spheres arranged in Cubic array

2.3.3 Fermat array

The most likely arrangement of spheres is a Fermat array. A saturated Fermat array is on display at a local supermarket in the form of a pile of oranges. Six spheres sit touching next to each other, as shown in Figure 2.4. The centers of the spheres correspond with the vertices of a hexagon. A seventh sphere fits in the middle of the hexagon. The next layer of spheres is identical but shifted to occupy the valleys between the spheres in the layer immediately below.

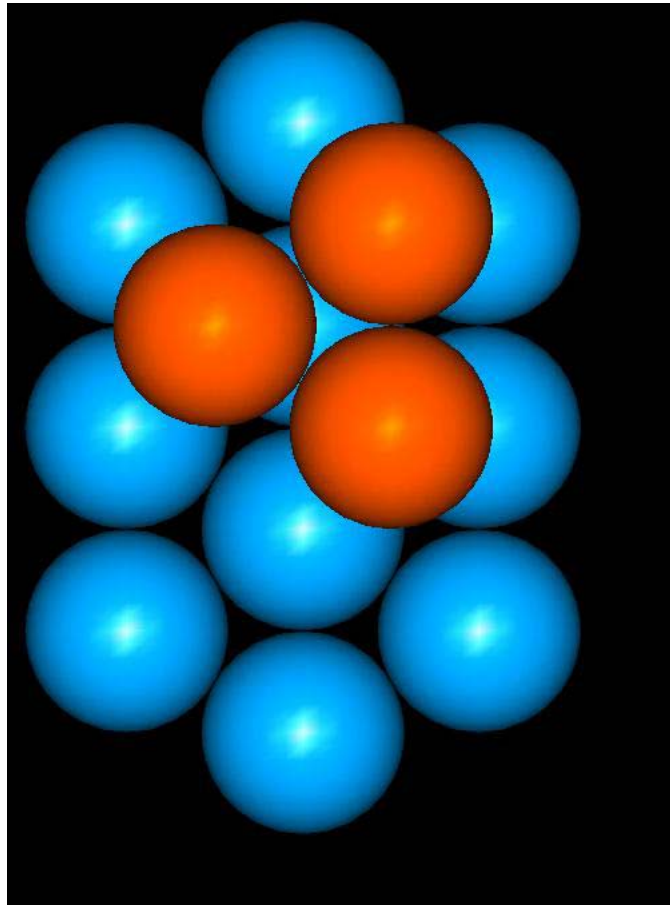


Figure 2.3.3 Fermat array

In the model, the array will not be saturated. This type of micro-structure is generated by creating a Fermat grid with all the possible positions. Then, the random generator is used to pick positions until sufficient spheres are placed to achieve the target porosity (Fig 2.5). The Fermat grid does allow for overlap in the same way as in the cubic array, i.e. by generating a grid tighter than the one corresponding to touching spheres.

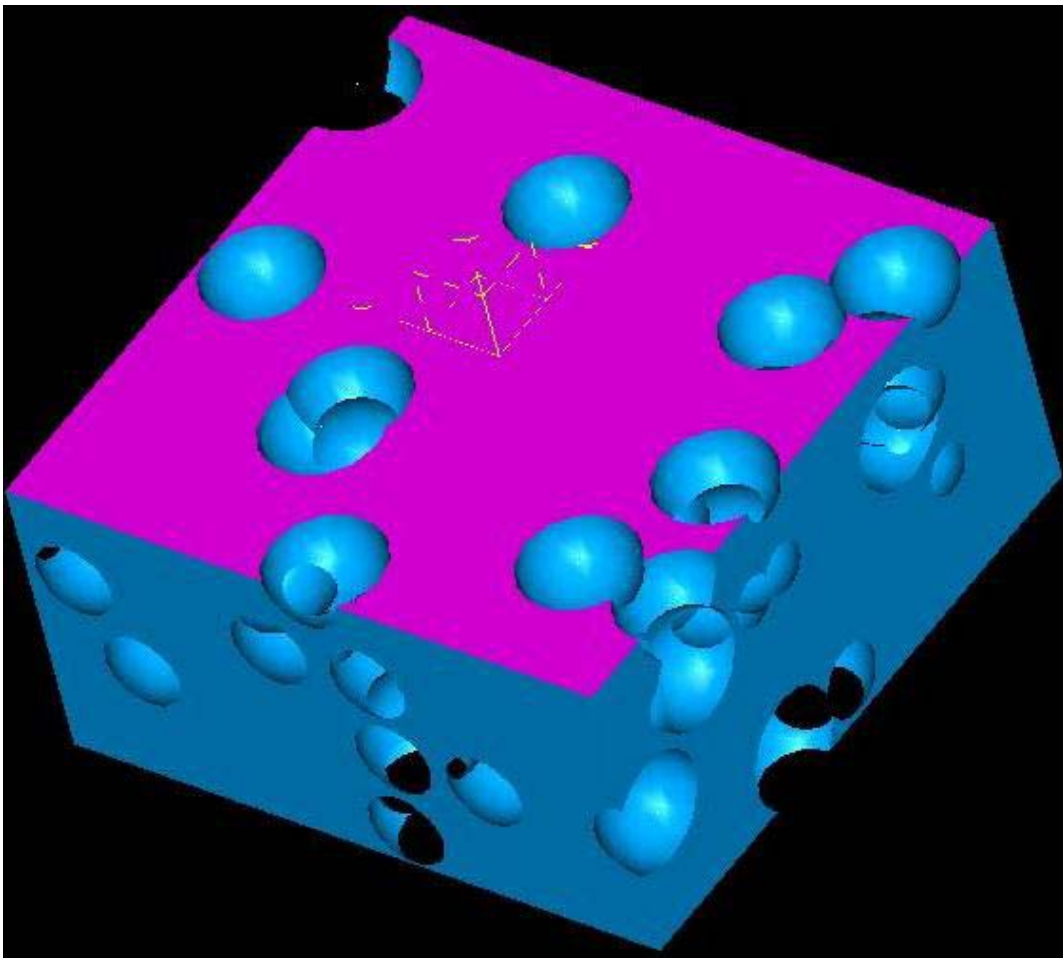


Figure 2.3.4 RVE with spheres arranged in Fermat array

2.4 Cylindrical voids

Compared to spheres, cylindrical voids have an additional dimension, the length l . They also add the complexity of orientation. Due to limitations of the mesh generation algorithms, time, and scope of the research, all models were limited to parallel cylinders with aspect ratio $l/d=2$.

Overlap is still defined in terms of the diameter, as for spheres. Cylinders overlap diametrically and longitudinally. Due to the longitudinal overlap and to the preferential orientation, cylindrical voids induce anisotropic permeability. That is, permeability in the direction of the axis of the cylinders is higher than in the radial direction. In the sequel, the direction of the axis of the cylinders is called *longitudinal*.

2.4.1 Random array

As with spheres, randomness is accomplished by placing cylinders one by one randomly. Those locations for which a cylinder would overlap more than the specified value, i.e. 10%, are rejected, and another random location is attempted. Inclusions are placed randomly in the entire volume of the RVE, even if a portion of the inclusion lies outside the RVE. Then, the protruding parts of the inclusions are removed, thus providing inlets and outlets for fluid flow (Fig 2.6).

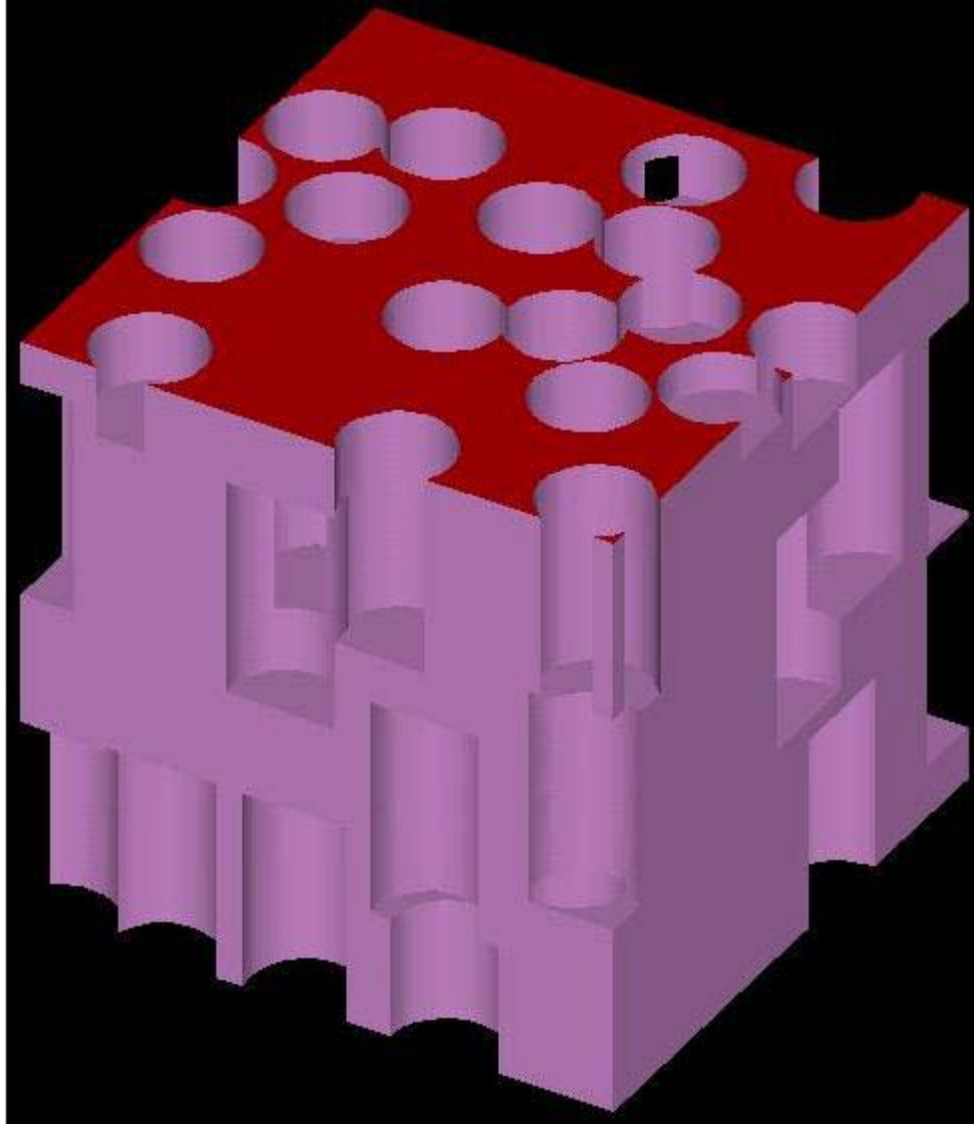


Figure 2.4.1 RVE with cylinders arranged in random array

2.4.2 Cubic array

As with spheres, the cubic array is a grid, which radially has spacing slightly lower than the diameter of the cylinders, to allow for overlap. In the longitudinal direction, the grid has spacing slightly shorter than the length of the cylinders. Inclusions are then placed randomly in the placeholders provided by the grid to achieve the target porosity (Fig 2.7).

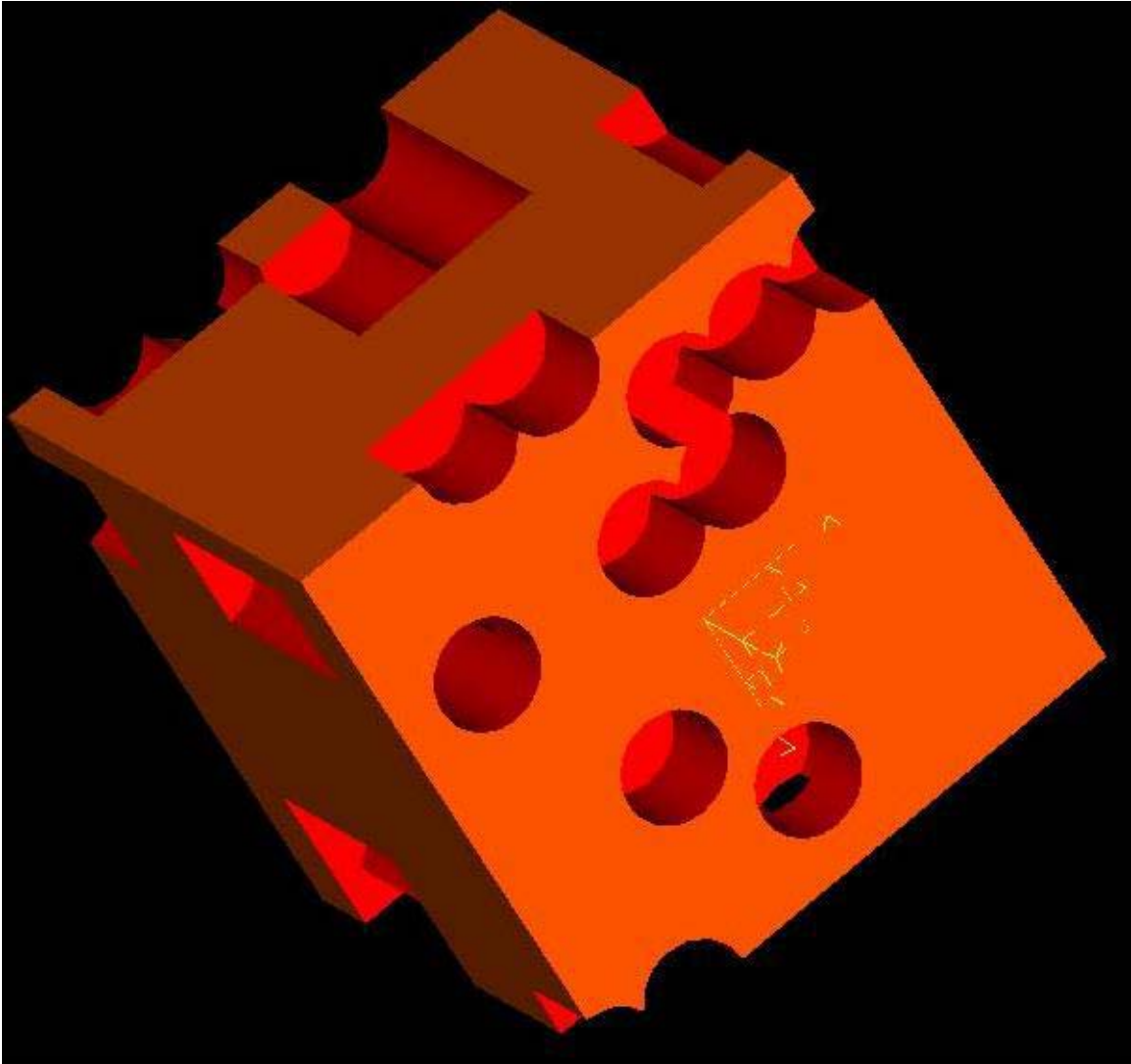


Figure 2.4.2 RVE with cylinders arranged in cubic array

2.4.3 Fermat array

Cylinders are placed in a two-dimensional Fermat grid, where the circular cross sections can be arranged in a Fermat configuration. The longitudinal direction is arranged in the same way of the cubic array. Of course overlap is allowed by making the grid slightly smaller than the dimensions of the cylinders. Inclusions are then placed randomly in the placeholders provided by the grid to achieve the target porosity.



Figure 2.4.3 RVE with cylinders arranged in Fermat array

The Marcos used to create these void arrangements are generated using FORTRAN[®] program. The FORTRAN[®] programs to generate random sphere and random cylinder are attached in Appendix I.

Chapter Three

Permeability

3.1 Introduction

Three types of permeability are defined as:

k_m , the permeability of the matrix,

k_v , the permeability of the material inside the voids,

and k_a , the apparent permeability of the resulting material.

Apparent permeability k_a is the average, overall, or macro permeability when the resulting material is considered as homogeneous, without paying attention to the micro-structure. The value of interest from the oil and gas production point of view is k_a . In Darcy permeability, k_m and k_v have finite values. Results are presented in terms of the permeability enhancement factor defined as k_a/k_m , which represents the factor by which the permeability of the matrix is enhanced by the presence of the voids.

3.2 Darcy Permeability

In order to study the permeability of a matrix with various arrangements of voids using a Darcy law model, it was necessary to assign a finite permeability value to the permeability of the void space. Initially, it was envisioned that the results for voids could be found using a large value of permeability for the voids, but not infinity. Such model

was verified numerically and it appeared that using a void/matrix permeability ratio $k_v/k_m=100$ produced virtually the same results as using $k_v/k_m \gg 100$.

In retrospect, the fallacy of the numerical models used to verify such assumption was that all models used were for relatively low porosity, in the range 0-20%. Such low porosity was purposely studied because of the intuitive concern that higher porosity would result in unacceptable high stress concentrations. For such low porosity, the micro-structural arrangements of the voids did not result in an interconnected path across which fluid flow could occur. Therefore, the apparent permeability k_a was controlled by the matrix permeability k_m , regardless of k_v .

The shortcomings of the Darcy model became apparent only after extensive study had taken place. Despite of the now obvious shortcomings of the Darcy model, the following tasks were accomplished:

All the micro-structural descriptions, both conceptual and computational, were developed in this phase of the study.

The void/matrix permeability ratio k_v/k_m has an effect on permeability enhancement k_a/k_m only when there is a clear path for fluid-flow. In other words, if the voids are not interconnected, it does not matter how high the permeability of the void material is, the value of k_a is controlled by the value of matrix permeability k_m .

The geometry of the microstructure is crucial, as it will become apparent in the results.

Before the appearance of a fluid-flow path, k_a grows slowly with porosity. Predicted permeability enhancement k_a/k_m is shown in Figure 3.1 for a cubic array of cylindrical voids. The permeability inside the voids was 100 times k_m . The voids have a diameter $d=640$ microns, and the ratio of void diameter d to RVE dimension a is $d/a=0.25$. The Darcy code in IDEAS™ is used to calculate the total fluid flow Q over the area A of one face of the RVE (Fig 3.1) when subjected to a pressure differential Δp . The result is used to compute the apparent permeability as

$$k_a = \mu L \frac{Q}{A \Delta p} \quad (1)$$

where Q is the fluid flow, A is the area of the face of the RVE, and Δp is the pressure drop from inlet to outlet, L is the length of the RVE in the direction of flow and μ is the viscosity (Fig 3.1). Before the appearance of a path, the apparent permeability results are not affected significantly by the ratio k_v/k_m . That is, k_m dominates the results (This feature can be observed till porosity of the model reaches 16%). After a path appears (i.e. at 24 %), the apparent permeability grows rapidly with the void porosity k_v , as it can be seen in Figure 3.2 for a cubic array of cylindrical voids. At 16% porosity, there is no path, and k_a is virtually constant. Above 24%, there is a path, and k_a grows with increasing k_v . When a fluid flow path is available, the apparent permeability is very high. This is confirmed in Section 3.3. As a result of this finding, emphasis of the project was shifted to finding the threshold minimum value of porosity (volume fraction of voids) that will ensure a 100% probability of creating a path. A fluid flow path appears at 24% porosity. At 45% porosity, k_a/k_m is only 20 in Figure 3.1 (when using $k_v/k_m = 100$). In

Figure 3.2, $\log(k_a/k_m)$ grow proportionally to $\log(k_v/k_m)$, which means that once there is a path for fluid flow, the apparent permeability can be very high. Better estimates of the actual values can be found by the CFD analysis. For each data point in the graphs, the path was identified visually by looking at the mesh in IDEAS™.

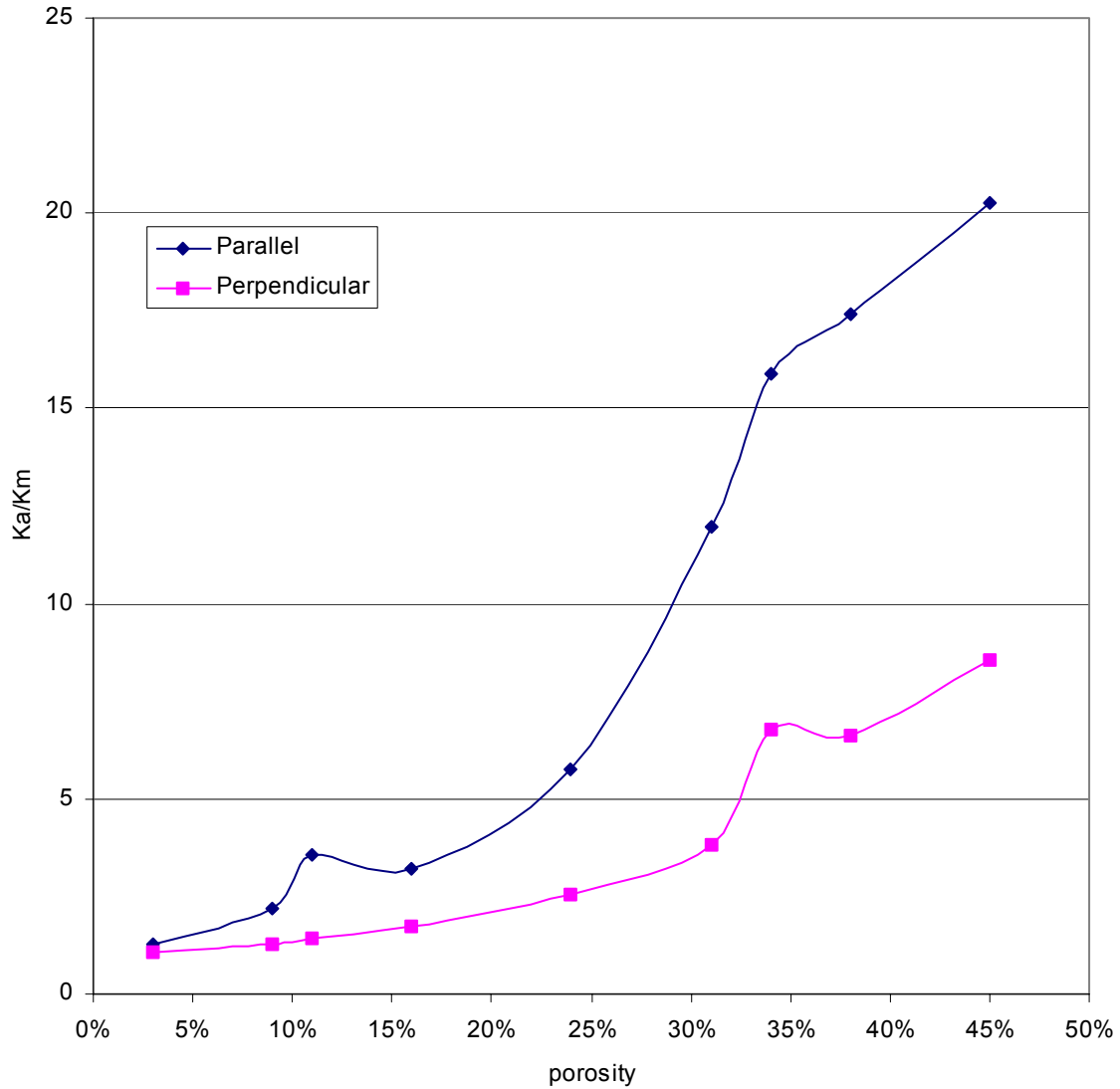


Figure 3.2.1 Permeability enhancement k_a/k_m of cubic array of cylinders with $k_v=100*k_m$.

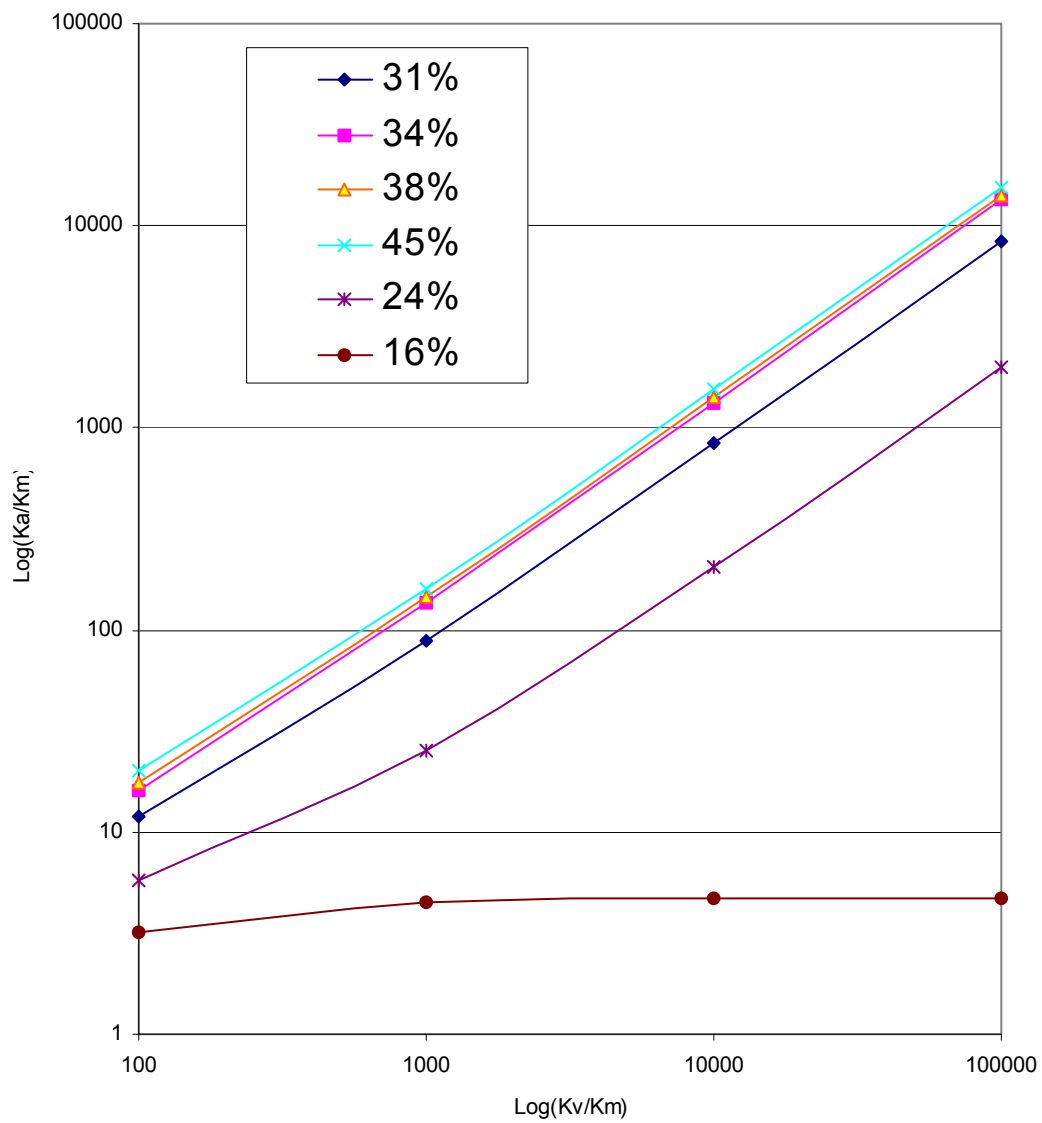


Figure 3.2.2. Apparent permeability enhancement k_a/k_m vs. dimensionless void permeability k_v/k_m .

3.3 CFD Permeability

It is apparent from previous chapter that a clear path is necessary to achieve high permeability. Therefore, it becomes necessary to study the flow through porous media where the pore space was empty and fully connected. In this section, $k_m=0$ and $k_v=\text{infinity}$; that is, the matrix is impervious, and the voids are empty.

To study flow through porous media with empty and fully connected pore space, we used FLUENT™, which is a commercial computational fluid dynamics code. Mesh generation was accomplished with a combination of a tailor-made FORTRAN code and GAMBIT™, the later being a commercial mesh generation program. The procedure to create a model using GAMBIT™ and solving it by FLUENT™ is described in Appendix III.

The simulation procedure is illustrated in Figure 3.3. The FORTRAN program is described in Appendix A. GAMBIT™ is then used to mesh the volume surrounding the voids. The procedure used to generate the mesh is described in Appendix C. The model is solved with FLUENT™.

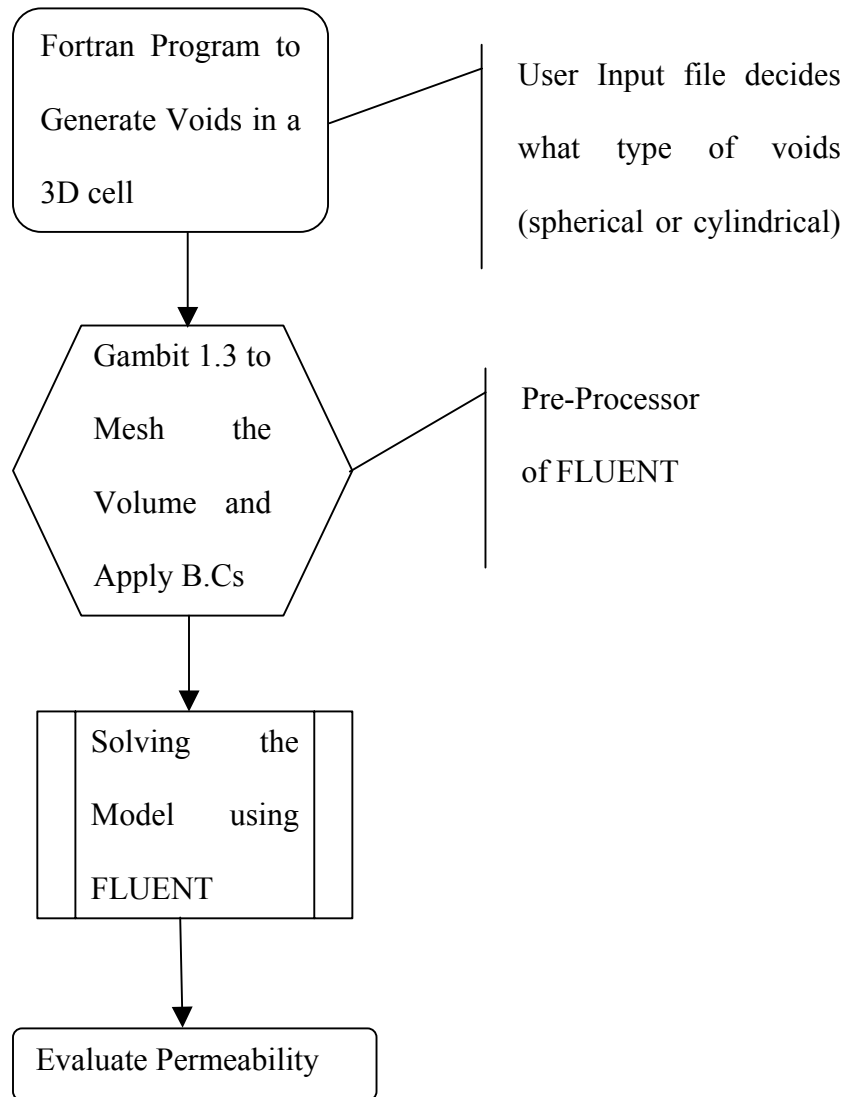


Figure 3.3.1 - CFD procedure.

For the simulations we used the following parameters¹: Density $\rho=850 \text{ Kg/m}^3$, viscosity $\mu=5.0 \cdot 10^{-3} \text{ N-s/m}^2$. The apparent permeability is computed from equation (2)

$$k_a = \mu L \frac{Q}{A \Delta p} \quad (2)$$

where A is the area of the face of the RVE, Q is the volumetric flow rate, L is the length of the RVE parallel to the flow direction, and Δp is the average static pressure drop, which is obtained from the results (see Figure 3.4). Since $Q=A_v \cdot V_{av}$, where V_{av} is the average velocity, which can be specified in FLUENTTM, we can use

$$k_a = \mu L \frac{V_{av} A_v}{A \Delta p} \quad (3)$$

The shortcoming of this approach is that it requires computation of the inlet flow area A_v through the face of the RVE, which is cumbersome to obtain for each model. Since the mass flow rate $m=\rho Q$ can be specified in FluentTM instead of the average velocity, and the resulting pressure drop can be obtained from the results (see Figure 3.5), we can use

$$k_a = \mu L \frac{m / \rho}{A \Delta p} \quad (4)$$

¹ Permeability: $m^2 = 1.013 \cdot 10^{15} \text{ mDarcy}$

Viscosity: $(\text{N-s/m}=\text{Kg/m/s})=100 \text{ (g/cm/s=cp)}$

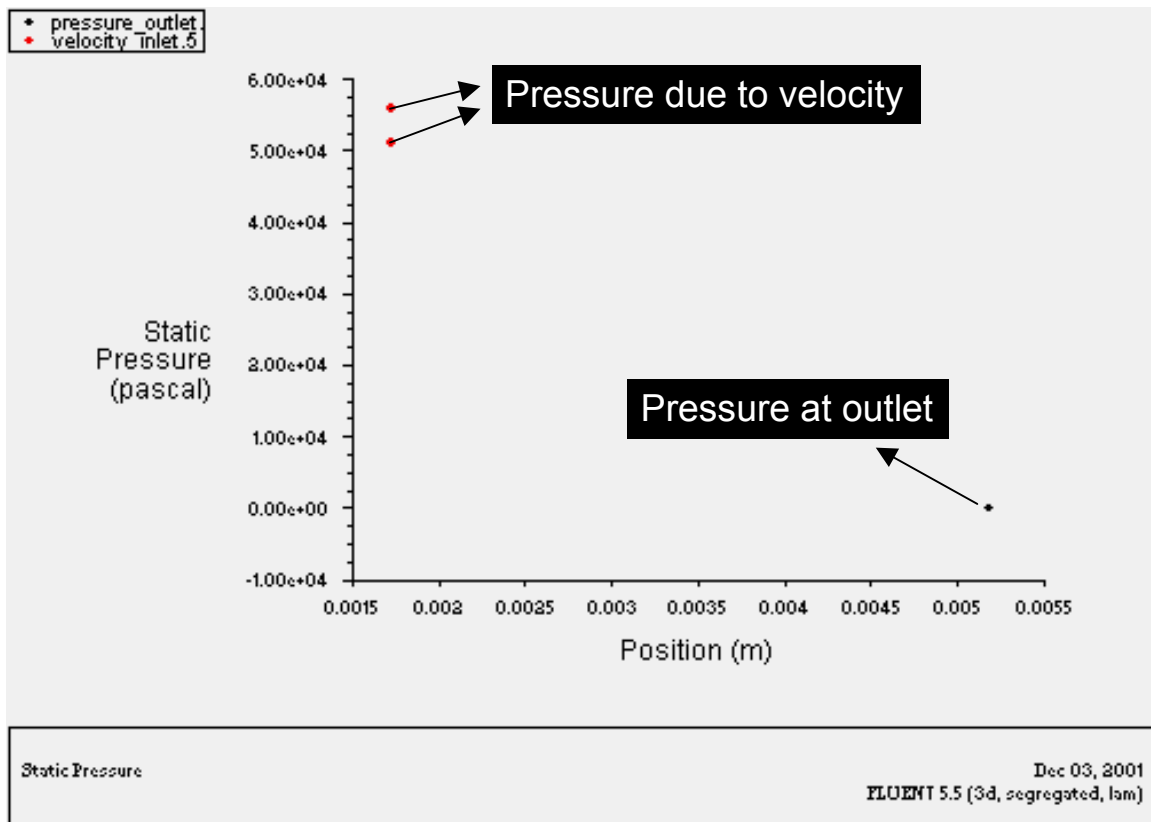


Figure 3.3.2 - Pressure drop when the inlet velocity is specified.

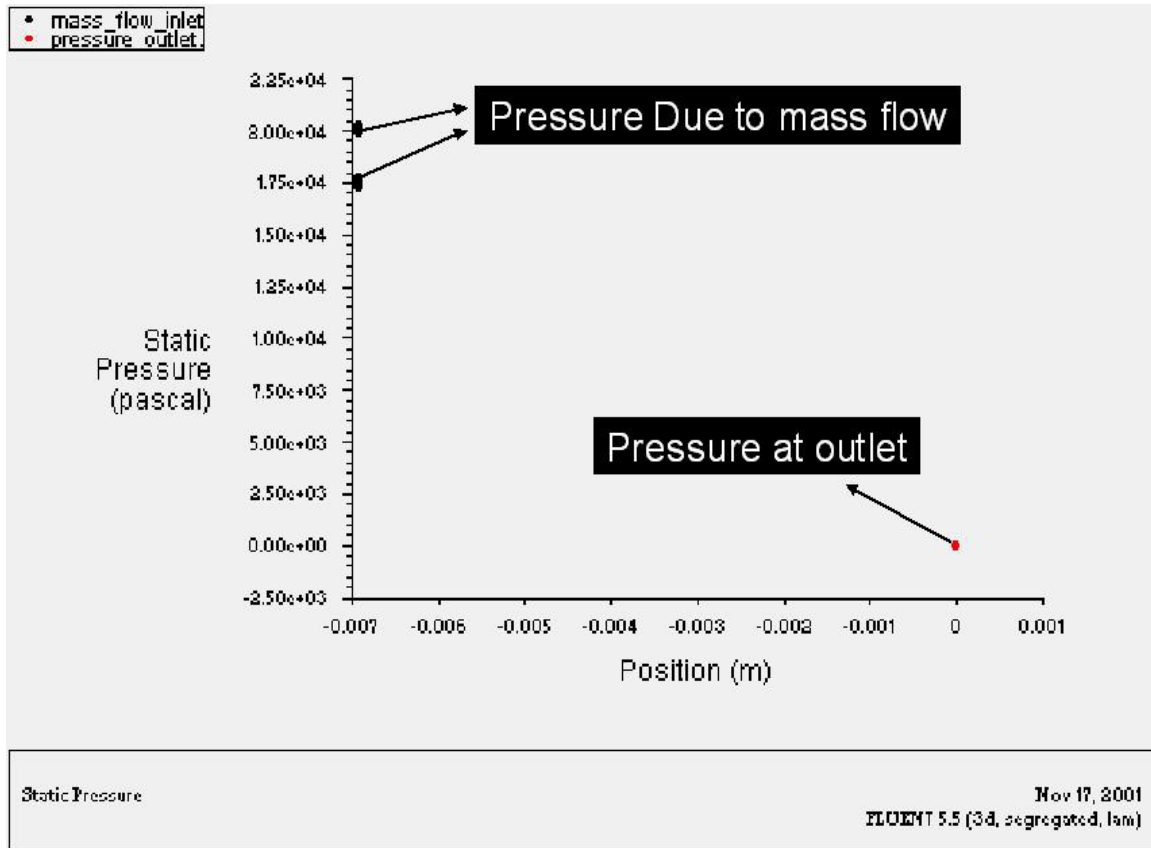


Figure 3.3.3 - Pressure drop when the inlet mass flow rate is specified.

Reynolds number is used to assess laminar vs. turbulent flow inside the void space

$$R_e = \frac{\rho V r}{\mu} \quad (5)$$

where V is the mean velocity, and r is a characteristic length, taken in this study as the diameter of the void. All computations were performed for laminar flow defined as $Re < 2300$. The procedure to calculate ΔP is described in appendix III.

3.4 Verification

Verification of the CFD simulation was performed by modeling 20-40 mesh sand with porosity 35%. A cubic array of spheres was used to simulate the sand as shown in Figure 3.6, where the inlet and outlet are shown. Only 1/8 of the RVE is actually meshed by taking advantage of symmetry, which results in substantial saving in computer time.

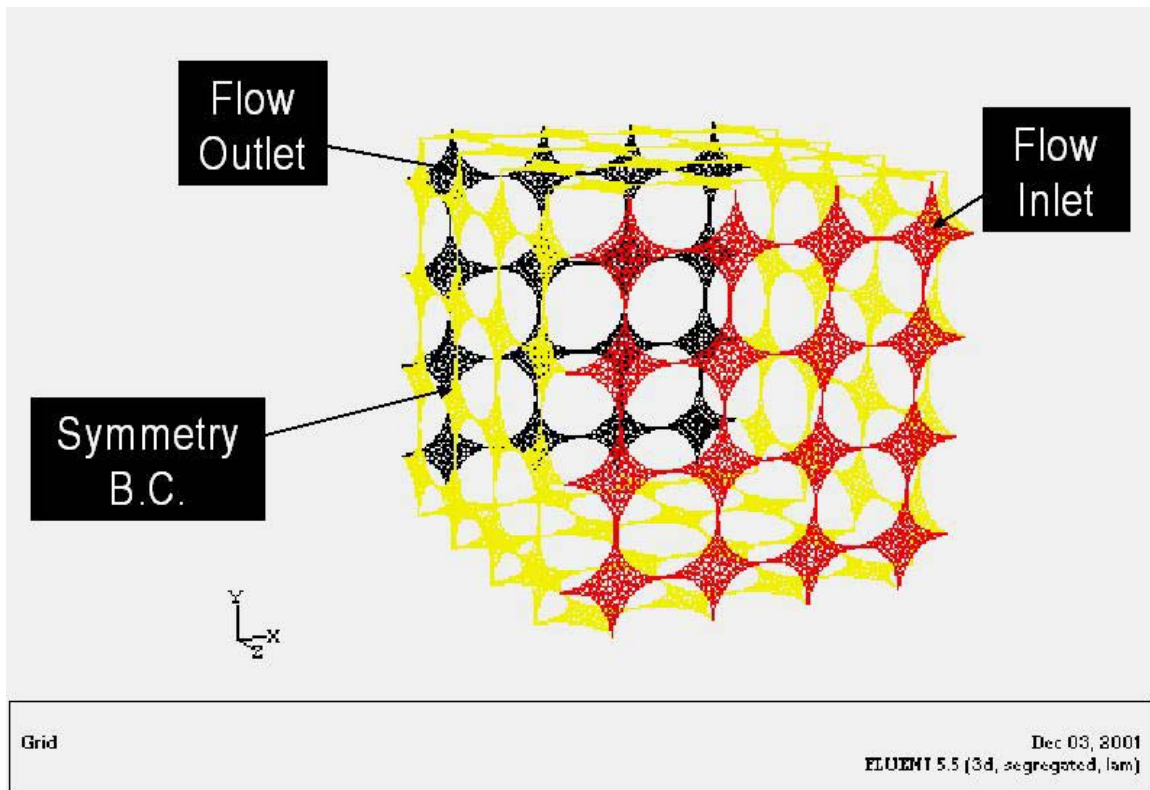


Figure 3.4.1 - Cubic array of sand (spherical) particles to simulate permeability of 20-40 mesh sand.

The direction of flow is along the Z-axis from the surface shown in red to the surface shown in black.

The resulting velocity contours are shown in Figure 3.7.

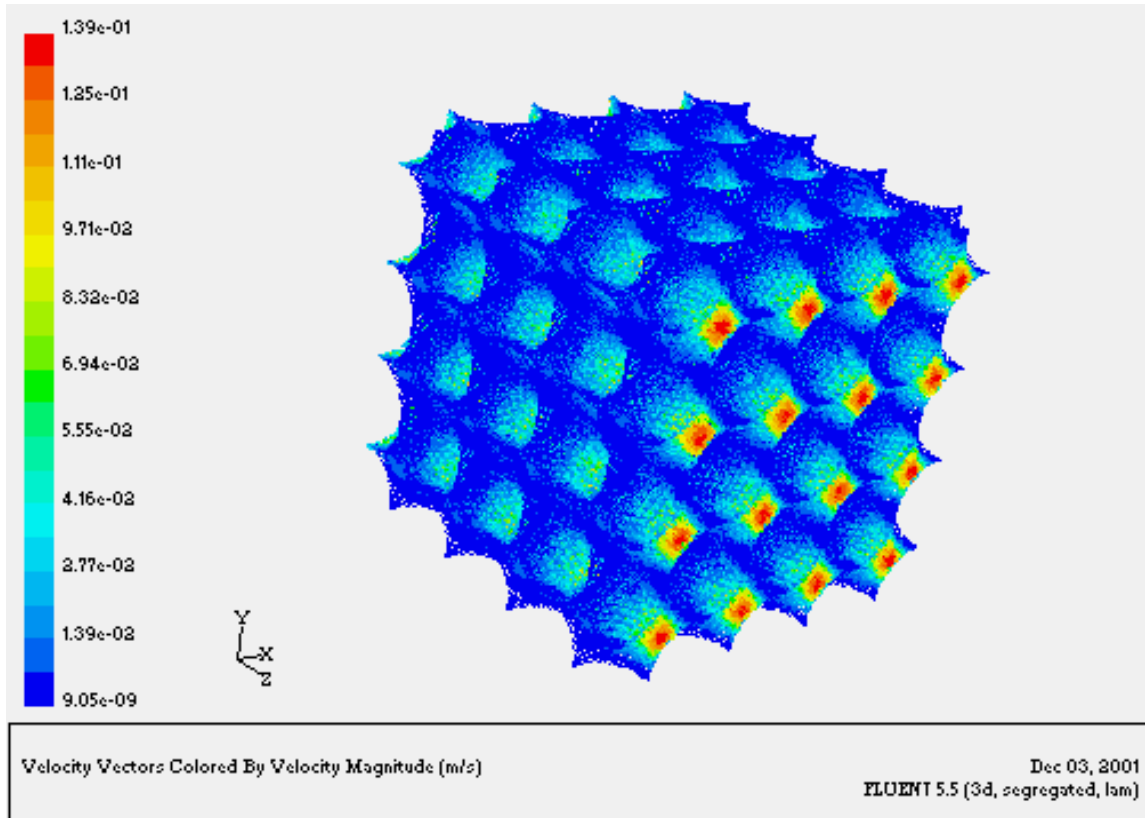


Figure 3.4.2 - Velocity contour in 20-40 sand model.

Next, the pressure drop between inlet and outlet is shown in Figure 3.5 for locations on the inlet and outlet face only. Using the pressure drop from this plot and the specified mass flow rate, we compute with equation (4) the apparent permeability of the modeled sand as $k=368 \mu\text{m}^2$. The experimental value is $k=119 \mu\text{m}^2$ for real sand. The difference is due to the fact that real sand does not arrange itself in a cubic array. Furthermore, the grain geometry is not spherical, there is no grain-to-grain separation, and the fluid viscosity used in the experiment was unknown. Despite the difference, we considered the model to be successful at predicting an approximate value of apparent permeability.

3.5 Threshold porosity for creating a flow path

Since the predicted permeability of microstructures with flow path was high, emphasis was shifted to the determination of the threshold porosity that would assure a fluid path for every void geometry and packing array.

We generated several models to determine the minimum volume fraction that yields a clear connected path from one face of the representative volume element (RVE) to the opposite. Porosity is the ratio of inclusion volume over RVE volume. Volume fraction is equal to the resulting porosity of the final artificially porous material. Not all pore space is connected to exterior faces. So not all pore space is effective.

For inlet and outlet we picked the top and bottom faces. For Spheres this was good enough because of their symmetry. In case of cylinders, a path was searched for in the transverse direction too.

We placed the following inclusions:

Spheres, diameter $d=640$ microns

Cylinders, diameter $d=640$ microns, aspect ratio length over diameter $L/d=2$.

The RVE has dimensions $2a \times 2a \times 2a$, where $a=3 \cdot (1-\text{overlap}) \cdot d$. overlap is in the range 0-1, from no overlap to fully overlapped inclusions. Overlap used in the model was 0.1

Placement took place on three geometries:

Cubic grid

Fermat grid

Random placement

Random placement (c) is completely random. Cubic & Fermat grids provide uniformly spaced placeholders for inclusions but not all placeholders are filled. Placement of an inclusion in any position in the grid is random. No two inclusions can occupy the same position. The grids provide possible locations for the inclusions. Grid size must fit an inclusion minus the overlap “o”. For spheres, grid size in all three directions x, y, z, is

$$S_x = S_y = S_z = d - o \quad (6)$$

For cylinders, the grid size across the diameter is same as for spheres,

$$S_x = S_y = d - o \quad (7)$$

but across the length L is

$$S_z = L - o \quad (8)$$

Since the RVE is not infinitely large, random placement does not produce a path every single time a model is generated. Instead, for a given volume fraction, there is a probability of finding a path in the RVE.

We generated five models at each value of volume fraction. In Tables 1-4, NP means no path and P means a path was found. We confirmed that a path exists by visually looking at the solid model in Gambit. The porosity had to be computed in each case for the actual model. This is because of the part of the inclusions (spheres, cylinders) which are partially outside the RVE would not count toward the actual porosity. Models had to be created in both IDEAS and Gambit. The models in Gambit would confirm the existence of path & the model in IDEAS was used to compute the exact porosity.

The threshold porosity at which we have 100% probability of finding a path (5/5) are given in Table 3.1.

Table 3.5.1. Threshold porosity and for each void arrangement.

Inclusion	Array	Direction	% porosity for 100 % probability of path
Spherical	Cubic	any	30%
Spherical	Fermat	any	30%
Spherical	Random	any	50%(45%)
Cylindrical	Cubic	Longitudinal	25%
Cylindrical	Cubic	Longitudinal	25%
Cylindrical	Fermat	Transverse	30%
Cylindrical	Fermat	Longitudinal	40%
Cylindrical	Random	Transverse	50%
Cylindrical	Random	Longitudinal	50%

Tables 3.5.2 through 3.5.10 show the values of porosity needed to achieve a path in all five cases.

**Table 3.5.2 Sphere
Random**

	Run Number				
15%	NP(14)	NP(14)	NP(14.5)	NP(14)	NP(15)
20%	NP(19)	NP(19)	NP(19.5)	NP(19)	NP(20)
25%	NP(23)	NP(23.3)	NP(24)	NP(23)	NP(24.5)
30%	NP(28)	NP(28)	NP(28.5)	NP(28)	NP(29)
35%	NP(32)	NP(32.4)	NP(33)	NP(32.1)	NP(32.7)
37%	NP(36.5)	NP(36.5)	NP(37.1)	NP(36.4)	NP(37.2)
40%	P(40.1)	NP(40.3)	NP(40.5)	NP(40.2)	NP(40.9)
45%	P(44.3)	NP(44.4)	NP(44.4)	NP(44.5)	P(45.4)
50%	P(47.4)	P(47.7)	P(48.5)	P(48.1)	P(49)

**Table 3.5.3 Sphere
Cubic**

	Run Number				
Vf	1	2	3	4	5
15%	NP(15.3)	NP(15.3)	NP(15.3)	NP(15.4)	NP(15.3)
20%	NP(20.7)	NP(20.6)	NP(20.5)	NP(20.7)	NP(20.7)
25%	P(26)	NP(25.7)	P(25.7)	NP(26)	NP(25.9)
30%	P(30.9)	P(30.7)	P(30.6)	P(30.9)	P(31)

**Table 3.5.4 Sphere
Fermat**

	Run Number				
Vf	1	2	3	4	5
15%	NP(15.3)	NP(15.3)	NP(15.3)	NP(15.4)	NP(15.3)
20%	NP(20.7)	NP(20.6)	NP(20.5)	NP(20.7)	NP(20.7)
25%	P(26)	NP(25.7)	P(25.7)	NP(26)	NP(25.9)
30%	P(30.9)	P(30.7)	P(30.6)	P(30.9)	P(31)

**Table 3.5.5 Cylinder Random
Radial**

	Run Number				
Vf	1	2	3	4	5
15%	NP(14.4)	NP(14.6)	NP(13.7)	NP(14.9)	NP(14.5)
20%	NP(18.9)	NP(18.8)	NP(18.5)	NP(19.6)	NP(19.4)
25%	NP(23.3)	NP(23.1)	NP(22.9)	NP(23.7)	NP(24.2)
30%	NP(27.7)	NP(27.1)	NP(26.9)	NP(27.4)	NP(28.6)
32%	NP(32.1)	NP(32)	NP(31.1)	NP(31.5)	NP(32.9)
35%	NP(35.9)	NP(35.6)	NP(35)	NP(36.1)	NP(36.4)
40%	NP(40.1)	NP(40)	P(39.9)	P(39.8)	NP(40.5)
45%	P(43.9)	NP(43.5)	P(42.6)	P(43.6)	NP(44.2)
50%	P(47.6)	P(46.8)	P(46.2)	P(46.8)	P(48.7)

**Table 3.5.6 Cylinder Random
Transverse**

	Run Number				
Vf	1	2	3	4	5
15%	NP(14.4)	NP(14.6)	NP(13.7)	NP(14.9)	NP(14.5)
20%	NP(18.9)	NP(18.8)	NP(18.5)	NP(19.6)	NP(19.4)
25%	NP(23.3)	NP(23.1)	NP(22.9)	NP(23.7)	NP(24.2)
30%	NP(27.7)	NP(27.1)	NP(26.9)	NP(27.4)	NP(28.6)
32%	NP(32.1)	NP(32)	NP(31.1)	NP(31.5)	NP(32.9)
35%	NP(35.9)	NP(35.6)	NP(35)	NP(36.1)	NP(36.4)
40%	NP(40.1)	NP(40)	P(39.9)	NP(39.8)	NP(40.5)
45%	NP(43.9)	NP(43.5)	P(42.6)	P(43.6)	P(44.2)
50%	P(47.6)	P(46.8)	P(46.2)	P(46.8)	P(48.7)

**Table 3.5.7 Cylinder Cubic
Radial**

	Run Number				
Vf	1	2	3	4	5
12%	NP(11.6)	NP(9.7)	NP(11.6)	NP(11.5)	NP(11.5)
15%	NP(14.3)	NP(14.2)	NP(15.5)	NP(14.7)	P(16.1)
20%	NP(18.2)	P(17.8)	P(19.6)	P(18.9)	P(19)
25%	P(23.3)	P(22)	P(23.3)	P(22)	P(24.7)

**Table 3.5.8 Cylinder Cubic
Transverse**

	Run Number				
Vf	1	2	3	4	5
12%	NP(11.6)	NP(9.7)	NP(11.6)	NP(11.5)	NP(11.5)
15%	P(14.3)	NP(14.2)	NP(15.5)	NP(14.7)	NP(16.1)
20%	P(18.2)	NP(17.8)	P(19.6)	P(18.9)	NP(19)
25%	P(23.3)	P(22)	P(23.3)	P(22)	P(24.7)

**Table 3.5.9 Cylinder Fermat
Radial**

	Run Number				
Vf	1	2	3	4	5
12%	NP(11.4)	NP(12.3)	NP(11.4)	NP(11)	NP(10.9)
15%	NP(15.3)	NP(15.8)	NP(15.3)	NP(14.8)	NP(14.3)
20%	NP(19.1)	P(19.8)	NP(19.8)	NP(18.2)	P(18.2)
23%	NP(22.8)	P(23)	NP(24)	NP(22)	P(22)
25%	NP(26.1)	P(26.9)	P(26.3)	NP(26.1)	P(26.1)
30%	P(29.3)	P(30.7)	P(30.1)	P(30)	P(28.9)
40%	P(37.9)	P(41)	P(40)	P(39.2)	P(39.7)

**Table 3.5.10 Cylinder Fermat
Transverse**

	Run Number				
Vf	1	2	3	4	5
12%	NP(11.4)	NP(12.3)	NP(11.4)	NP(11)	NP(10.9)
15%	NP(15.3)	NP(15.8)	NP(15.3)	NP(14.8)	NP(14.3)
20%	NP(19.1)	NP(19.8)	NP(19.8)	NP(18.2)	NP(18.2)
23%	NP(22.8)	NP(23)	P(24)	P(22)	NP(22)
25%	NP(26.1)	P(26.9)	P(26.3)	P(26.1)	P(26.1)
30%	NP(29.3)	P(30.7)	P(30.1)	P(30)	P(28.9)
40%	P(37.9)	P(41)	P(40)	P(39.2)	P(39.7)

3.6 Conclusions

The Darcy method was ineffective because the value of k_a was dominated by k_m until a clear path was formed. Moreover, the value of k_a will not rise as expected after a clear path is formed due to the finite value of k_v . In reality k_v would have a value of infinity, but for SDRC[®] IDEAS[®] to solve the model an infinite value cannot be attributed to k_v .

In order to replicate a realistic random distribution of the inclusions, a FORTRAN[®] program was developed to generate the locations of the inclusions. The FORTRAN[®] program generated the required microstructure, for spheres and cylinders in random, cubic, and Fermat arrays and the output was a macro that could be run in Gambit[™] as well as SDRC IDEAS[™]. Gambit[™] was successfully used to generate the mesh for the fluid space. It was also used to determine visually if fluid path was created by the intersection of the voids. If a path is created, then Fluent[™] was used to predict the flow and pressure drop, from which the apparent permeability was calculated using a MathCad[®] program. Since the predicted permeability of microstructures with flow path was high, emphasis was shifted to the determination of the threshold porosity that would assure a fluid path for every void geometry and packing array. The results show that the random packing arrays need a high porosity to create a path. Cubic array had path even at low porosity. The more ordered the packing, the less porosity is needed to create a path.

The permeability was calculated from the results obtained from solving the models using Fluent[®]. The permeability values were computed for only one configuration (40 % Fermat array with cylindrical inclusions). The pressure drop across the inlet and outlet

was determined using the graph similar to the one shown in Figure 3.5. Δp calculated from the graph is used input for the MathCAD program. The value of k_a was calculated using the MathCAD program. Since no experimental data exists for this material the results were compared with commercially made metallic foams. The k_a value from the commercial metallic foam called Duocel foam [11] was calculated for 40% porosity. It was found to be $13.88\mu\text{m}^2$ [11]. The results for models of 40% porosity are shown in table 3.11.

Table 3.6.1 k_a values of models with cylindrical voids of porosity 40% having Fermat arrangement.

Arrangement	milli Darcy	μm^2
C1	130	0.12833
C2	280	0.276403
C3	17	0.016782
C4	17	0.016782
C5	47	0.046396

The disparity in the results can be attributed to bottle necks and lack of conducting channels in the models.

Chapter Four

Stress Concentration

4.1 Introduction

Stress concentration factor (SCF) is defined as the ratio of the maximum stress experienced to the average stress in terms of the equivalent von Misses stress.

$$SCF = \frac{\sigma_{eq}}{p}$$
$$\sigma_{eq} = \sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_1 - \sigma_3)^2} \quad (1)$$

σ_i : principal stresses
p: applied pressure

In terms of the maximum principal stress, SCF is defined as

$$SCF = \frac{\max(\sigma_i)}{p} \quad (2)$$

Stress concentration increases rapidly with porosity as shown in Fig. 4.1-4.2. The values depend on the particular arrangement of the voids in the microstructure, and no two models are alike. The procedure to create a model to perform stress concentration analysis using SDRC-IDEAS is given in Appendix II

4.2 Stress Concentration

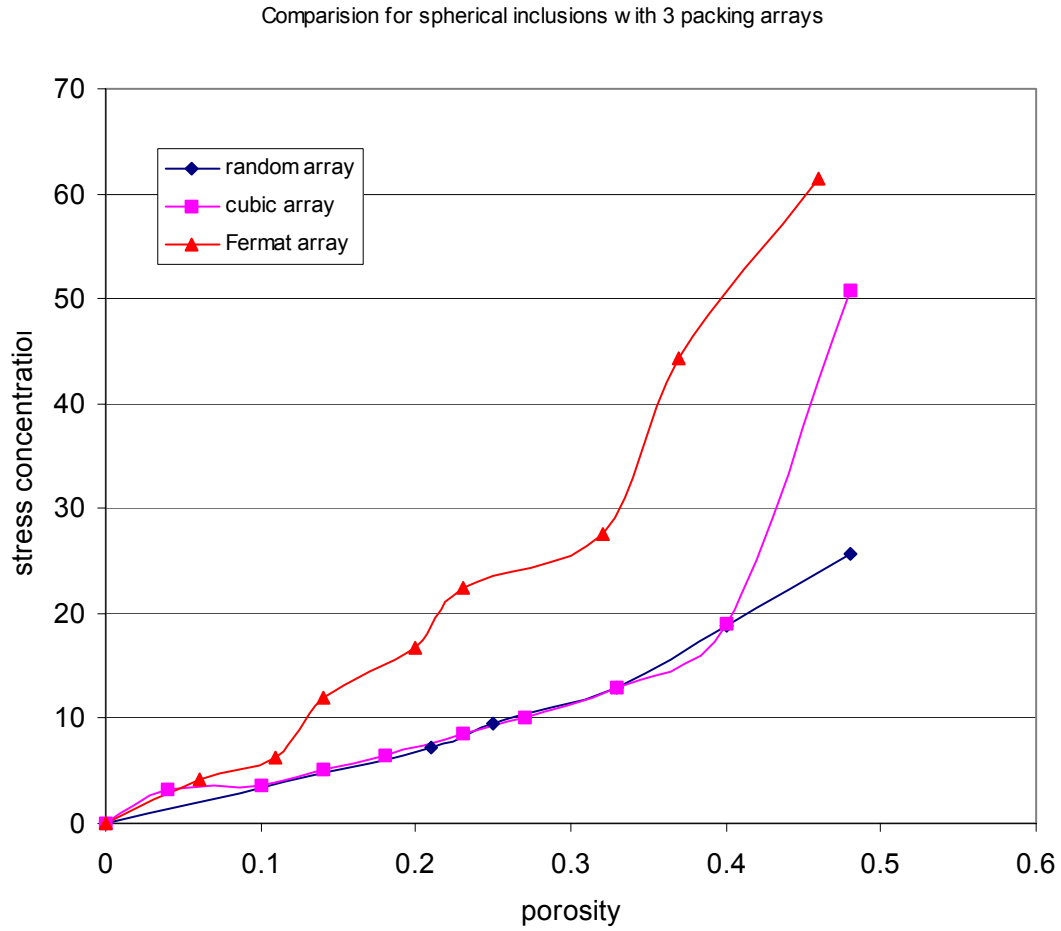


Figure 4.2.1 - Stress concentration factor vs. porosity for spherical inclusions.

The stress concentrations in Fermat array and cubic array appear to be higher than that of random array. This attributed to a thin sliver of material left in the block which is causing the variation in stress concentration.

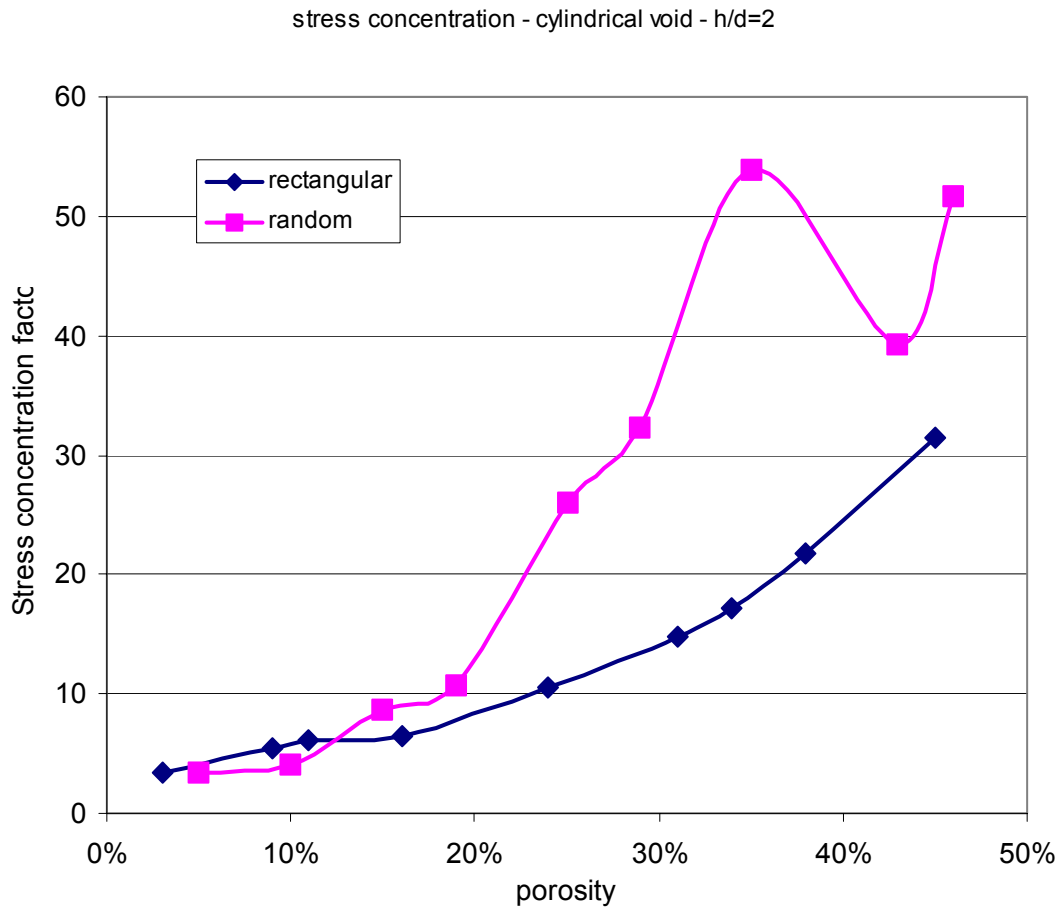


Figure 4.2.2 - Stress concentration factor vs. porosity for cylindrical inclusions, $l/d=2$.

While generating the new locations for spheres at 43%, the locations up to 35% are unchanged, the FORTRAN[®] program adds new voids to the already existing voids. So the sliver of material that was causing higher stress concentration was removed. This lead to drop in stress concentration from 35% porosity to 43% porosity.

The threshold void fraction required to achieve connectivity from one side of the unit cell to another was determined as described in chapter 3. In this section, stress analysis was performed for all the six configurations at threshold void-fraction to find the stress concentrations. Both hydrostatic and uniaxial loadings were considered. The stress

concentrations for each run (C1 to C5), Average, and Standard Deviation, are shown below. NM means no-mesh; that is when the software is unable to mesh the volume because of the complexity of the microstructure.

Table 4.2.1 Stress concentration factors for Cylindrical voids arranged Randomly @ 50% porosity

Run Number	Hydrostatic	Uniaxial
C1	423.3	285.186
C2	92.5	44.512
C3	17	54.596
C5	49.8	31.088
C4	NM	NM
Average	138.8	90.67
SD	187.7	121.276

Table 4.2.2 Stress concentration factors for Cylindrical voids arranged in cubic array @ 25% porosity

Run Number	Hydrostatic	Uniaxial
C1	NM	NM
C2	10.5	7.387
C3	10.4	8.8
C4	9.6	7.772
C5	12.3	8.222
Average	10.7	8.045
SD	1.2	0.608

Table 4.2.3 Stress concentration factors for Cylindrical voids arranged in Fermat array @ 30% porosity

Run Number	Hydrostatic	Uniaxial
C1	26.7	9.313
C2	22.4	13.553
C3	25.9	15.03
C4	20	10.341
C5	34.6	11.497
Average	25.9	11.947
SD	5.5	2.335

Table 4.2.4 Stress concentration factors for spherical voids arranged Randomly @ 45% porosity

Run Number	Hydrostatic	Uniaxial
C1	NM	NM
C2	10.5	7.387
C3	10.4	8.8
C4	9.6	7.772
C5	12.3	8.222
Average	10.7	8.045
SD	1.2	0.608

Table 4.2.5 Stress concentration factors for spherical voids arranged in cubic array @ 30% porosity

Run Number	Hydrostatic	Uniaxial
C1	10.9	12.140
C2	10.3	11.819
C3	10.7	13.810
C4	10	12.525
C5	10.5	14.837
Average	10.5	13.026
SD	0.4	1.264

Table 4.2.6 Stress concentration factors for spherical voids arranged in ferret array @ 40% porosity

Run Number	Hydrostatic	Uniaxial
C1	26.3	32.244
C2	21.6	56.395
C3	21.1	24.986
C4	36.7	33.721
C5	27.8	39.117
Average	26.7	37.293
SD	6.3	11.81

Using IDEAS[®] we couldn't mesh sphere random at 50% void fraction for 4 of the 5 runs due to the complexity of the microstructure. We could mesh 45% for three models (reported in Table 4.4), and 50% for one model (reported in Figure 4.1).

We observed an interesting phenomenon while doing the stress analysis for sphere random in run number C2. The stress concentration actually dropped when the volume fraction increased from 45% to 50% as shown in Figure 4.3. While generating the new locations for spheres at 50%, the locations up to 45% are unchanged from run C2 at 45%. So, the new spheres are cut at new locations from the 45% model. In other words, the model adds new voids to the already existing voids. For run C2, the new voids cut out material that had very high stress concentration, thus effectively reducing it. We think that modeling crumbling of the material will have an even more pronounced effect at lowering the stress concentration. We think so because the material that will crumble will be invariably that with highest stress concentration. Crumbling will reduce the stress concentrations from the values reported here.

The graph shows that stress concentration drops significantly at 50 % from the maximum at 45%. This is due to the fact that a volume, which had high stress concentration, was removed (serendipitously). Effectively, the stress concentration has dropped. In reality, what is likely to happen is that the material with high stress concentration will crumble, thus removing the high stress concentration volume.

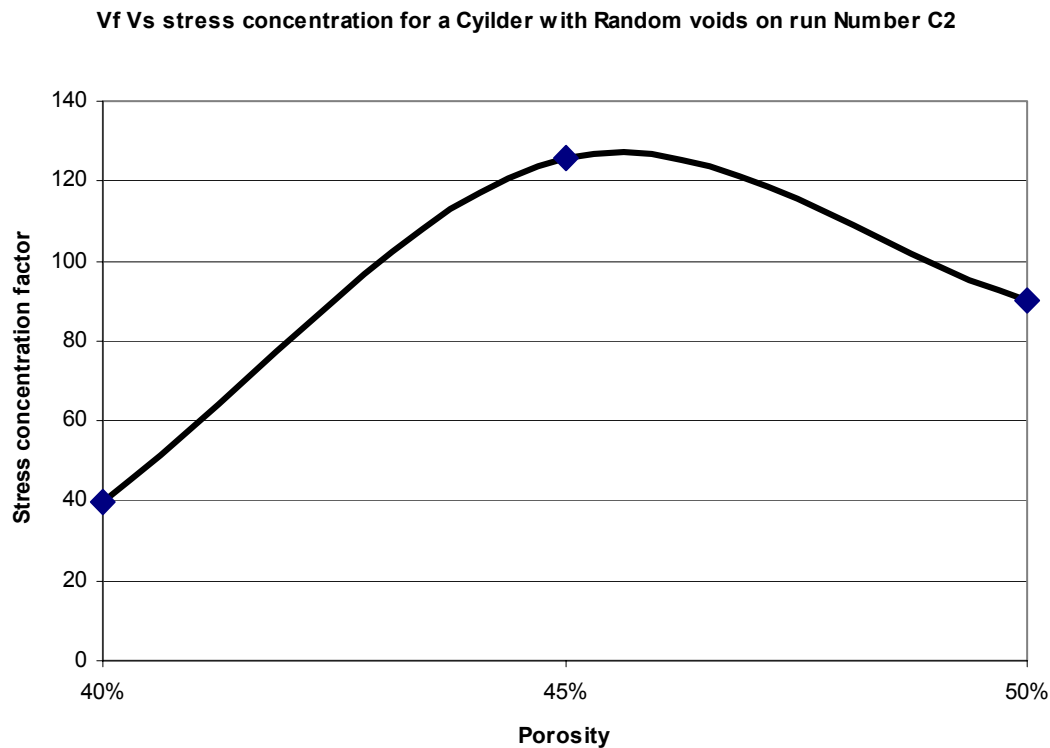


Figure 4.2.3 - Stress concentration factor vs. porosity for spheres with random packing.

4.3 Crumbling

The computation of the stress concentration factor assumes that the matrix is linearly elastic. High stress concentration can be encountered when a thin sheet or sliver of material is left after two or more voids are created closed to each other.

We attempted to model the effect that a yielding matrix would have on reducing the stress concentration, after the regions of high stress have yielded. Ideally, the stress would be redistributed to other areas that hopefully would be more massive and thus would have less stress concentration.

Because of limited time and heavy demands on computer resources, we could not solve actual models with plasticity. Instead, we illustrated the concept with a simpler model. The simplified model consists of two spherical voids barely touching each other near the center of the RVE (Fig. 4.4), leaving a thin sliver of material. The material is modeled as elastic-plastic. The plastic behavior was assumed to be bi-linear with an elastic modulus $E=3.9017E10$ Pa, Poisson's ratio $\nu=0.18$, tangent modulus after yield assumed to be $E_T=E/100$ and yield strength assumed to be $\sigma_y=E/100$ as well.

The model was validated by computing the elastic stress concentration factor, then recalculating it with the non-linear code, using AnsysTM, to make sure the value was identical. After that, sufficient hydrostatic pressure was applied to reach and exceed the yield point.

It can be seen in Figure 4.5 that the SCF (calculated as the von Mises stress over the applied pressure) is initially high as long as the material remains elastic. Then it drops when yielding occurs. The volume of material that yields gives an indication of the amount of crumbling that can be expected. It can be seen that significant drop of SCF occurs with small crumbling. For a 4% volume crumbled, SCF drops from 4.0 to 1.0 (von Mises). For this example, the minimum value of SCF occurs with approximately 18% crumbling. This crumbling further increases the porosity and might help in achieving higher permeability.

The von Mises stress is an equivalent stress that measures the amount of shear stress in the material. The drop in von Mises stress means that yielding the material at those locations of high stress is a very effective mechanism to redistribute and lower shear stress.

The SCF calculated as the ratio between the maximum principal stress over the applied pressure also drops significantly in accordance with the SCF calculated from von Mises stress. For a 4% volume crumbled, SCF drops from 3.3 to 1.5 (principal stress). This proves the argument that when the material yields, it lowers the stress

concentrations, because the volumes that caused the high stress concentrations would have yielded or crumbled.

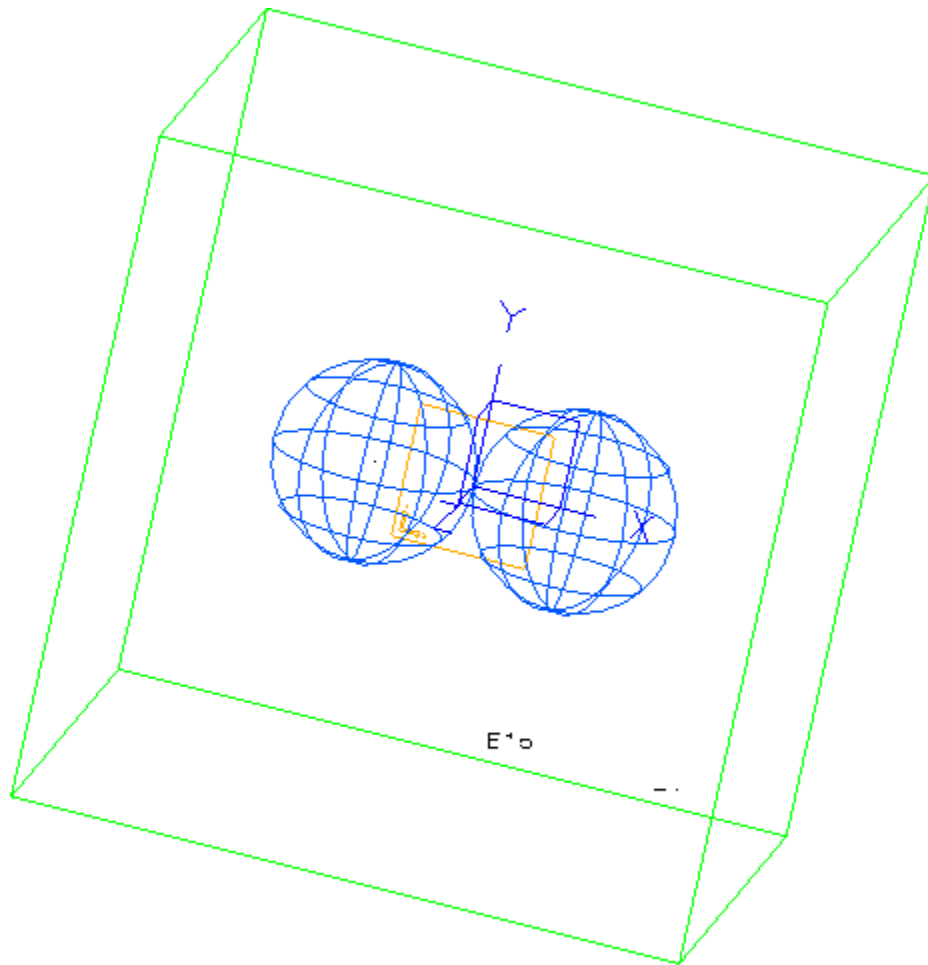


Figure 4.3.1 - Two-void model used to study crumbling.

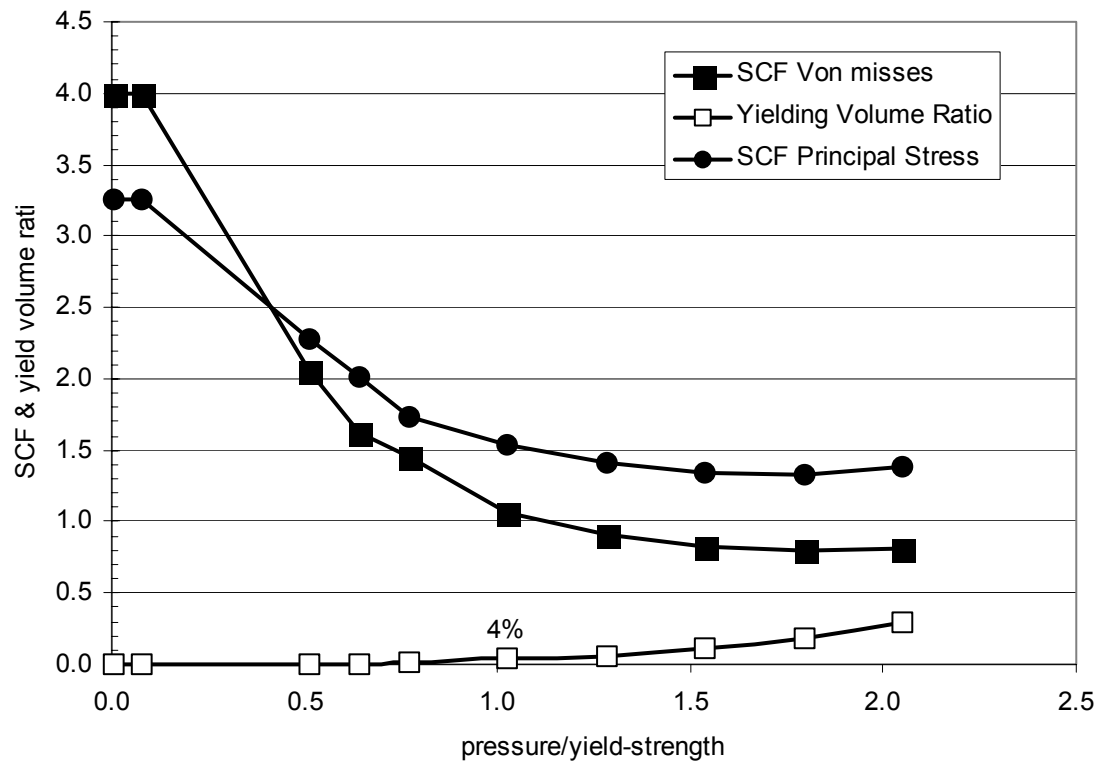


Figure 4.3.2 - Effect of yield on stress concentration factor (SCF).

Chapter Five

Conclusions and Recommendations

5.1 Conclusions

The main objective of this research was to design a material that can be used by Halliburton Company as proppant for hydraulic fracturing. The introduction of voids enhances the permeability. An initial design using cube shaped voids was inferior due to high stress concentrations. A spherical void was effective in reduction of stress concentration but higher permeability values were attained when cylindrical voids. It was realized that existence of a path was the deciding factor in the permeability enhancement.

Initially it was thought random arrangement of voids was a realistic arrangement. But this idea was abandoned in favor of Fermat arrangement. Fermat array is a more likely arrangement. When two voids are close by they will not be separate but will come close to each other due to surface tension, forming a peanut shaped void. This feature could not be simulated in random arrangement.

The RVE could not be modeled with pressure loading since it resulted in rigid body motion. So the RVE was loaded on one face corresponding opposite face was restrained in the direction of loading.

Darcy permeability had its drawback. The permeability of the void had to be a finite value so that the model could be solved. But in reality the permeability of a void is

infinity. So this approach had to be abandoned. The results from Darcy permeability indicated the importance of existence of a path.

Analysis in FluentTM and GambitTM with the principles of Computational fluid dynamics gave a better approximation of overall permeability of the material.

Cubic array has the highest permeability and the lowest stress concentration. But this only a theoretical arrangement and would not occur in reality. Cylinders had better permeability and lower stress concentrations because they form channels easily and have better load bearing capabilities.

Halliburton was looking for high permeability with low stress concentrations. The analysis performed showed that to achieve higher permeability the porosity has to be around 35 to 40%. At these porosities the stress concentration values are very high. This would mean that the material would be crushed due to rock closure stresses. Halliburton withdrew this idea. The Finite element analysis performed saved Halliburton lot money and time, since the idea seemed unworthy of pursuing.

5.2 Recommendations

Recommendations to refine and obtain better approximation of stress concentration are as follows:

- Perform more analysis to find optimum radius of a spherical void that result in lower stress concentrations and higher permeability.

- Consider analyzing models using different radii for the cylindrical voids and different aspect ratios (d/l).
- Perform analysis on matrix containing cylindrical voids with random orientations.
- Consider performing crumbling on all models to lower the effective stress concentration

References

- 1) Matt Bayfield. S.L. Fisher and L.M. King “Burst and collapse of a sealed multilateral junction : Numerical simulation” Spe drilling & completion journal Volume 15, number 2 75-83.
- 2) Mahmoud Asadi, Mogdeh K. Shirazi, Ali Ghalambor, and Dan W. Pratt “Experimental and simulation analysis of Jet-perforated rock damage” Spe drilling & completion journal Volume 16, number 3 176-181.
- 3) W.W. Aud, T.D. Poulson, R.A. Burns, T.R. Rushing, and W.D. Orr “Lateral proppant distribution: the good, the bad, and the ugly of putting the frac jobs away” Spe drilling & completion journal Volume 16, number 1 4-11.
- 4) J.D. Engel “Application of the “Superfrac” process in Mid-Continent area”. American Petroleum Institute Drilling and production practice Journal 213- 219.
- 5) Huiping Ma and Douglas Ruth “ Physical explanations of Non-Darcy effects for fluid flow in Porous Media” SPE formation evaluation Journal volume 12 number 1 13-18.
- 6) Robert S. Schechter “Oil well Stimulation”.
- 7) J. S. Archer and C. G. Wall “Petroleum Engineering- Principles and Practice”.
- 8) Martinez, S. et al “Formation Fracturing” paper SPE series 28.
- 9) Michael J. Economides and Kenneth G. Nolte “Reservoir stimulation” second edition.
- 10) Kern, L.R., and Perkins, T.K., Trans. AIMME, 216 (1959) 403.
- 11) <http://ergaerospace.com/flowtest.htm>
- 12) SDRC-IDEAS manual.

13) Gambit[®] help files.

14) Fluent[®] help files.

15) Ansys[®] manual.

Appendix I

I.1 Program for generating Random Sphere

Program new

```
real(8) x(10000),y(10000),z(10000),a(3),f,dd1,dd2,over,pi,r
real(8) dx,dy,dz,dxyz(10000)
integer factor,key,type1,dir,counter,nv,i,j,k,c
OPEN (11,file='c:\matlab\bin\pream\input1.txt')

read(11,*) Key, type1, dir
read(11,*) f
read(11,*) dd1, dd2, over

pi=3.14159265359d0

r=dd1/2

factor=3
a(1) = factor*(1e0-over)*dd1;
a(2) = a(1); a(3) = a(1) !the dimensions of the cube are decided here

open (62,file='c:\temp.jou.txt')
write(62,15)'volume create "cube" width ',2*a(1),' depth ',
12*a(2),' height ',2*a(3),' brick'
write(62,16)'volume move "cube" offset ',0,0,0
write(62,15)'volume create "ball" radius 0.32e-0 sphere'

nv1 = (8*f*a(1)*a(2)*a(3))/(4*pi*(r)**3/3))
nv=1.1*nv1

counter=0
dist=(1e0-over)*dd1

i=1
j=1
loop1: do while (i<=nv)
j=j+1
if(i>10000) then
write(*,*) "Exceeded dimension limit on x,y,z"
goto 1000
endif
c1=j**2 ; c2=j**5;c3=j**3;c4=2*j**4;c5=j**6
c=c2
```

```

r1=ran(c)
r1=ran(c)
r2=ran(c)
r3=ran(c)
x(i)=2e0*a(1)*r1-a(1)
y(i)=2e0*a(2)*r2-a(2)
z(i)=2e0*a(3)*r3-a(3)
!write(*,*) i, r1,r2,r3
c   write(*,*) i, x(i)
c   k=old
c   i=new
temp=1E12
loop2: do k=1,i
if (k/=i) then
dx= abs(x(i)-x(k))
dy= abs(y(i)-y(k))
dz= abs(z(i)-z(k))
dxyz(k)=sqrt(dx**2+dy**2+dz**2)
if(dxyz(k)<temp) temp=dxyz(k)
c   else
c   dxyz(k)=1E12
end if
end do loop2

write(*,*) i, temp,dist

if(temp>dist)then
write(62,11)'volume copy "ball" to "volume.',i,'"
write(62,12)'volume move "volume.',i,'" offset',x(i),y(i),
1z(i)
i=i+1
endif

end do loop1
write(*,*) " Tryed",j
!Creates Ideas File

open (40,file='c:\ideas.prg.txt')

write(40,14)"AP: 1 8 Change View"
write(40,14) "AP: 1 0 0 0 0"
write(40,14)"AP: 0.0      0.0      0.0"
write(40,14)"AP: 1.000000    0.0      0.0"
write(40,14)"AP: 0.0      1.000000    0.0"
write(40,14)"AP: 0.0      0.0      1.000000"

```

```

write(40,14)"AP: 0.1050000    0.2520000    0.2520000
    1    15.000000"
write(40,14)"AP: -1.000000    -1.000000    -1.000000"
write(40,14)"AP: 1.000000    1.000000    1.000000"
write(40,14)"AP: 2 11 Program File User Preferences"
write(40,14)"AP: 0 1 0 0 0 0 1 0"
write(40,14)"AP: 1 -0.1000000    -0.1000000    0.1000000
    1    0.1000000"
write(40,14)"AP: 2 0.0        0.0        0.0
    1        0.0"
write(40,14)"AP: 3                1
    10                0"
write(40,14)"AP: 4                0
    11                0"
write(40,14)"AP: 5                0
    10                1"
write(40,14)"AP: 6                0
    10                0"
    write(40,14)"AP: 7 0 0 0 10 10 0.05000000    0.05000000
    1    0.05000000    30.00000"
14 Format(TL1,A)

```

```

write(40,14)"AP: 8 0"
write(40,14)"AP: 9 1 1"
write(40,14)"AP: 10 1 0 0 20 2.000000"
write(40,14)"K : $ return"
write(40,14)"K : $ mpos ;; /ma ge"
write(40,14)"K : ET 2;"
write(40,14)"K : APPL"
write(40,14)"K : ET 2;"
write(40,14)"K : APPL"
write(40,14)"K : OKAY"
write(40,14)"K : $ return"
write(40,14)"K : $ REDI"
write(40,14)"AP: 1 8 Change View"
write(40,14)"AP: 1 0 0 0 0"
write(40,14)"AP: 0.0        0.0        0.0"
write(40,14)"AP: 1.000000    0.0        0.0"
write(40,14)"AP: 0.0        1.000000    0.0"
write(40,14)"AP: 0.0        0.0        1.000000"
write(40,14)"AP: 0.1050000    0.2767650    0.2767650
    1    15.00000"
write(40,14)"AP: -1.000000    -1.000000    -1.000000"
write(40,14)"AP: 1.000000    1.000000    1.000000"
write(40,14)"K : "

```

```
write(40,14)"K :"  
write(40,14)"K :"
```

```
Loop10: Do j = 1,nv  
  write(40,14)"K : $ /or mo"  
  write(40,14)"K : lab"  
  write(40,14)"K : ball"  
  write(40,14)"K : pt"  
  write(40,14)"K :"  
  write(40,14)"K : c"  
  write(40,14)"K : on"  
  write(40,2)"K : ",x(j),y(j),z(j)  
  write(40,14)"K : l"  
  write(40,14)"K : DON"  
  write(40,14)"K : $ mpos ;; /ma na"  
  write(40,14)"K : lab"  
  write(40,14)"K : untitled1"  
  write(40,6)"K : N ",j  
  write(40,14)"K : OKAY"  
  write(40,14)"K :"  
  write(40,14)"K : $ return"  
end do Loop10
```

```
2 Format(TL1,A,f9.5,1X,f9.5,1X,f9.5)
```

```
6 Format(TL1,A,I3.3)
```

```
  i3=001  
  write(40,14)"K : $ mpos ;; SAVE"  
  write(40,14)"K : $ return"  
  n = 0  
  Loop11: do j = 2,nv  
    n = n + 1  
    If (n == 21) Then  
      n = n - 20  
    End If  
    write(40,14)"K : $ /co j"  
    write(40,14)"K : lab"  
    write(40,7)"K : ",",",j,""  
    write(40,7)"K : ",",",i3,""  
    write(40,14)"K :"  
    If (n == 20) Then  
      write(40,14)"K : $ mpos ;; SAVE"  
      write(40,14)"K : $ return"  
    End If  
  
  end do Loop11
```

```

write(40,14)"K : $ mpos ;; SAVE"
write(40,14)"K : $ return"
write(40,14)"K : "
write(40,14)"E : **** END OF SESSION ****"

7 Format(TL1,A,A,I3.3,A)

15 Format(TL1,A,f9.5,A,f9.5,A,f9.5,A)
16 Format(TL1,A,f6.4,1X,f6.4,1X,f6.4)
11 Format(TL1,A,I3.1,A)
12 Format(TL1,A,I3.1,A,1X,f9.5,1X,f9.5,1X,f9.5)
1000 end program

```


I.2 Program for generating Random Cylinder

Program new

```
real(8) x(10000),y(10000),z(10000),a(3),f,dd1,dd2,over,pi,r
real(8) dx,dy,dz,dxyz(10000),dh(10000)
integer factor,key,type1,dir,counter,nv,i,j,k,c
OPEN (11,file='c:\matlab\bin\pream\input1.txt')

read(11,*) Key, type1, dir
read(11,*) f
read(11,*) dd1, dd2, over

pi=3.14159265359d0

r=dd1/2

factor=3
a(1) = factor*(1e0-over)*dd1;
a(2) = factor*(1e0-over)*dd2;
a(3) = a(1) !the dimensions of the cube are decided here

open (62,file='c:\temp.jou.txt')
write(62,15)'volume create "cube" width ',2*a(1),' depth ',
12*a(2),' height ',2*a(3),' brick'
write(62,16)'volume move "cube" offset ',0,0,0
!write(62,15)'volume create "ball" radius 0.32e-0 sphere'
write(62,17)'volume create "cylinder" height ', dd2,
1' radius1 ', r,' radius3 ', r,' yaxis frustum'
nv1 = (8*f*a(1)*a(2)*a(3)/(pi*dd2*(r)**2))
nv=1.1*nv1

counter=0
dist1=(1e0-over)*dd1
dist2=(1e0-over)*dd2

c   new proposed inclusion=i
    i=1
c   trying as much as it takes=j
    j=1
    loop1: do while (i<=nv)
        j=j+1
        if(i>10000) then
            write(*,*) "Exceeded dimension limit on x,y,z"
```

```

goto 1000
endif
c1=j**3 ; c2=j**4;c3=2*j**3;c4=2*j**5;c5=j**6
c=c3
r1=ran(c)
r1=ran(c)
r2=ran(c)
r3=ran(c)
x(i)=2e0*a(1)*r1-a(1)
y(i)=2e0*a(2)*r2-a(2)
z(i)=2e0*a(3)*r3-a(3)
!write(*,*) i, r1,r2,r3

c      distances between proposed inclusion and closest existing one
temp1=1E12
temp2=1E12

c      check new proposed inclusion against all previous ones=k
loop2: do k=1,i
if (k/=i) then
dx= abs(x(i)-x(k))
dy= abs(y(i)-y(k))
dz= abs(z(i)-z(k))
dxyz(k)=sqrt(dx**2+dz**2)
dh(k)=dy

if(dxyz(k)<dist1.and.dh(k)<dist2) goto 1001
c      if(dxyz(k)<temp1.and.dh(k)<temp2) then
c      temp1=dxyz(k)
c      temp2=dh(k)

end if
c      end if
c      write(*,*) i, dxyz(k),dh(k),temp1,temp2
end do loop2

c      write(*,*) i, temp1,temp2,dist1,dist2
c      if(temp1>dist1.and.temp2>dist2)then

write(62,11)'volume copy "cylinder" to "volume.',i,'"
write(62,12)'volume move "volume.',i,'" offset',x(i),y(i),
1z(i)
i=i+1
c      endif

```

1001 end do loop1

write(*,*) " Tried",j
!Creates Ideas File

open (40,file='c:\ideas.prg.txt')

write(40,14)"AP: 1 8 Change View"
write(40,14) "AP: 1 0 0 0 0"
write(40,14)"AP: 0.0 0.0 0.0"
write(40,14)"AP: 1.000000 0.0 0.0"
write(40,14)"AP: 0.0 1.000000 0.0"
write(40,14)"AP: 0.0 0.0 1.000000"
write(40,14)"AP: 0.1050000 0.2520000 0.2520000
1 15.000000"
write(40,14)"AP: -1.000000 -1.000000 -1.000000"
write(40,14)"AP: 1.000000 1.000000 1.000000"
write(40,14)"AP: 2 11 Program File User Preferences"
write(40,14)"AP: 0 1 0 0 0 0 1 0"
write(40,14)"AP: 1 -0.1000000 -0.1000000 0.1000000
1 0.1000000"
write(40,14)"AP: 2 0.0 0.0 0.0
1 0.0"
write(40,14)"AP: 3 1
10 0"
write(40,14)"AP: 4 0
11 0"
write(40,14)"AP: 5 0
10 1"
write(40,14)"AP: 6 0
10 0"
write(40,14)"AP: 7 0 0 0 10 10 0.05000000 0.05000000
1 0.05000000 30.000000"

14 Format(TL1,A)

write(40,14)"AP: 8 0"
write(40,14)"AP: 9 1 1"
write(40,14)"AP: 10 1 0 0 20 2.000000"
write(40,14)"K : \$ return"
write(40,14)"K : \$ mpos ;; /ma ge"
write(40,14)"K : ET 2;"
write(40,14)"K : APPL"
write(40,14)"K : ET 2;"
write(40,14)"K : APPL"
write(40,14)"K : OKAY"

```

write(40,14)"K : $ return"
write(40,14)"K : $ REDI"
write(40,14)"AP: 1 8 Change View"
write(40,14)"AP: 1 0 0 0 0"
write(40,14)"AP: 0.0      0.0      0.0"
write(40,14)"AP: 1.000000  0.0      0.0"
write(40,14)"AP: 0.0      1.000000  0.0"
write(40,14)"AP: 0.0      0.0      1.000000"
write(40,14)"AP: 0.1050000  0.2767650  0.2767650
      1  15.000000"
write(40,14)"AP: -1.000000  -1.000000  -1.000000"
write(40,14)"AP: 1.000000  1.000000  1.000000"
write(40,14)"K : "
write(40,14)"K : "
write(40,14)"K : "

```

```

Loop10: Do j = 1,nv
  write(40,14)"K : $ /or mo"
  write(40,14)"K : lab"
  write(40,14)"K : ball"
  write(40,14)"K : pt"
  write(40,14)"K : "
  write(40,14)"K : c"
  write(40,14)"K : on"
  write(40,2)"K : ",x(j),y(j),z(j)
  write(40,14)"K : l"
  write(40,14)"K : DON"
  write(40,14)"K : $ mpos ;; /ma na"
  write(40,14)"K : lab"
  write(40,14)"K : untitled1"
  write(40,6)"K : N ",j
  write(40,14)"K : OKAY"
  write(40,14)"K : "
  write(40,14)"K : $ return"
end do Loop10

```

```

2 Format(TL1,A,f9.5,1X,f9.5,1X,f9.5)

```

```

6 Format(TL1,A,I3.3)

```

```

      i3=001
  write(40,14)"K : $ mpos ;; SAVE"
  write(40,14)"K : $ return"
  n = 0
Loop11: do j = 2,nv
  n = n + 1
  If (n == 41) Then

```

```

        n = n - 40
    End If
    write(40,14)"K : $ /co j"
    write(40,14)"K : lab"
    write(40,7)"K : ", "", j, ""
    write(40,7)"K : ", "", i3, ""
    write(40,14)"K : "
    If (n == 20) Then
        write(40,14)"K : $ mpos ;; SAVE"
        write(40,14)"K : $ return"
    End If

end do Loop11

write(40,14)"K : $ mpos ;; SAVE"
write(40,14)"K : $ return"
write(40,14)"K : "
write(40,14)"E : **** END OF SESSION ****"

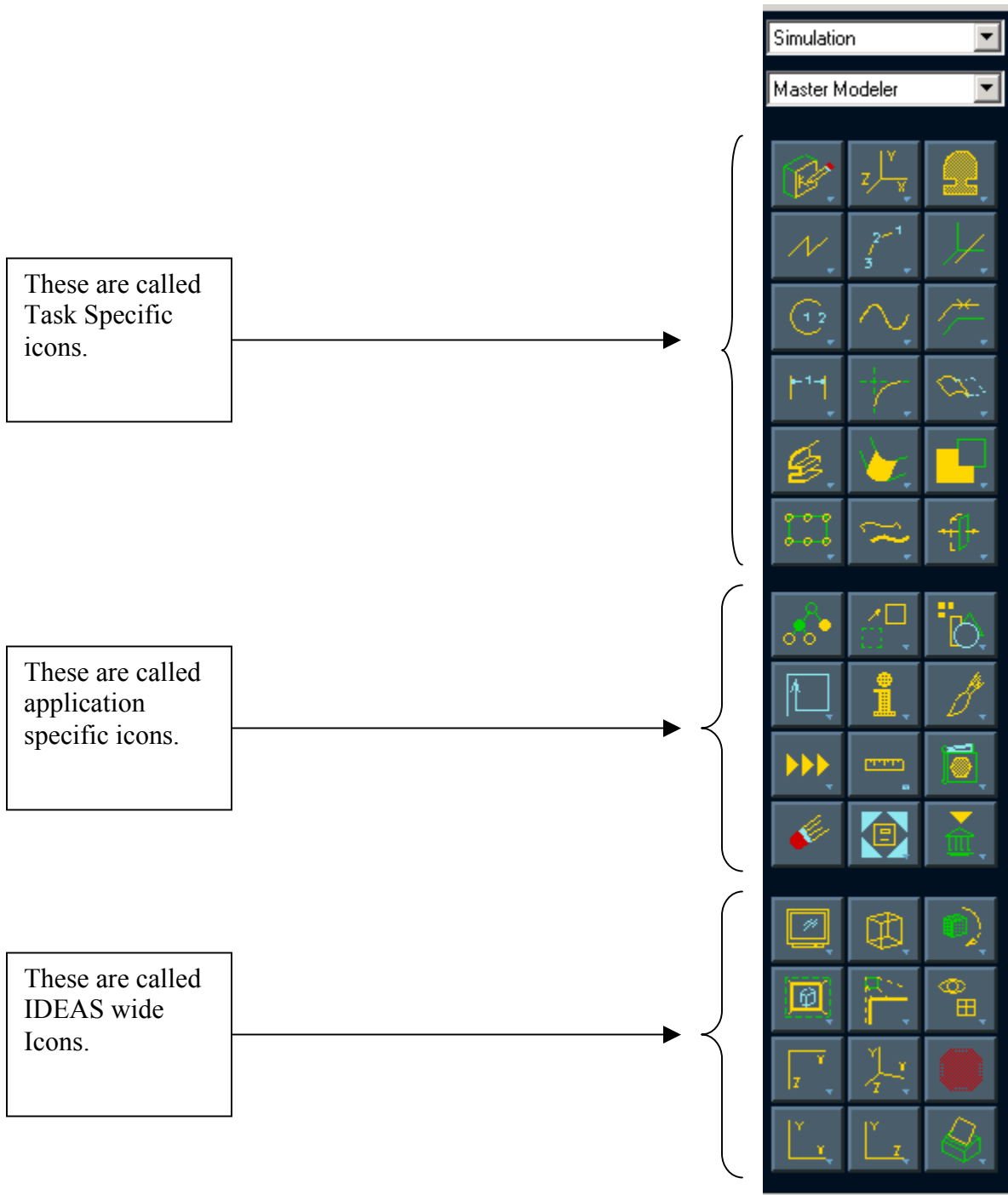
7 Format(TL1,A,A,I3.3,A)

17 Format(TL1,A,f6.4,A,f6.4,A,f6.4,A)
15 Format(TL1,A,f9.5,A,f9.5,A,f9.5,A)
16 Format(TL1,A,f6.4,1X,f6.4,1X,f6.4)
11 Format(TL1,A,I3.1,A)
12 Format(TL1,A,I3.1,A,1X,f9.5,1X,f9.5,1X,f9.5)
1000 end program

```

Appendix II

II.1 Creating a new model



There are also other function keys. F1, F2 and F3 help dynamic viewing. Sometimes there it's difficult to find what you want to pick. These keys help in changing the view.

Holding F1 and moving the mouse sideways or up and down will pan the picture depending on the motion of the mouse.

Holding F2 and moving the mouse up and down will magnify or shrink the picture. Moving the mouse up will shrink the picture and moving the mouse down will magnify the picture.

Holding F3 will help rotating the picture in X, Y or Z direction. If the cursor is within 80% of the viewport center, the mouse controls X and Y rotation. If the mouse is outside the 80% range the mouse controls Z rotation.

Holding F7 will perform “zoom all “.

Holding F8 will perform “reconsider “.

Holding F9 will perform “deselect all “.

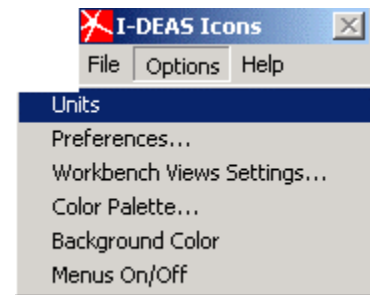
Holding F11 will perform “filter “.

Holding F12 will perform “redisplay “.

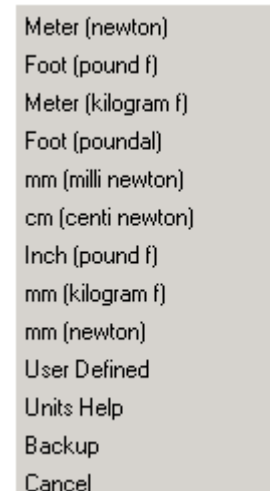
Open a new file in IDEAS. Choose master modeler module.




Click and hold the options menu. A menu opens as shown in the picture. Choose unit.



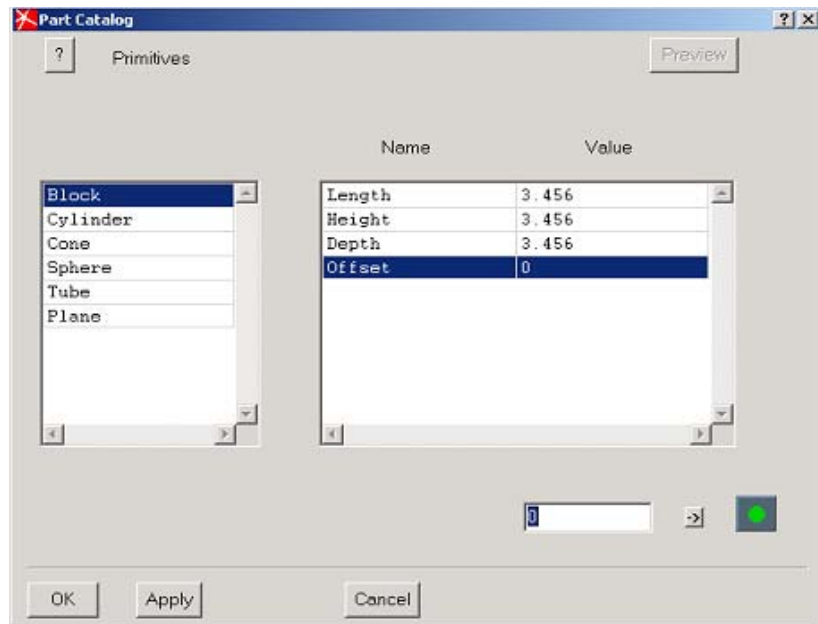
This opens a new menu on the main window.



Choose meter (newton) from the menu and press enter or the middle mouse button. Choosing this will set the units of all the values entered by the user and the constants used by the program to SI unit system.



Click and hold on  (parts) icon to create a new part. This opens a new menu. Choose the same icon.


This opens a new window. There are different kinds of preset shapes that can be generated. The only disadvantage is that complex shapes cannot be defined by this menu.






Click on block and choose the dimensions of block as shown and press enter.

Then click on sphere and set the radius as 0.32 and press enter. This creates a cube and a sphere whose centers are (0,0,0). The cube and the sphere have to be named.

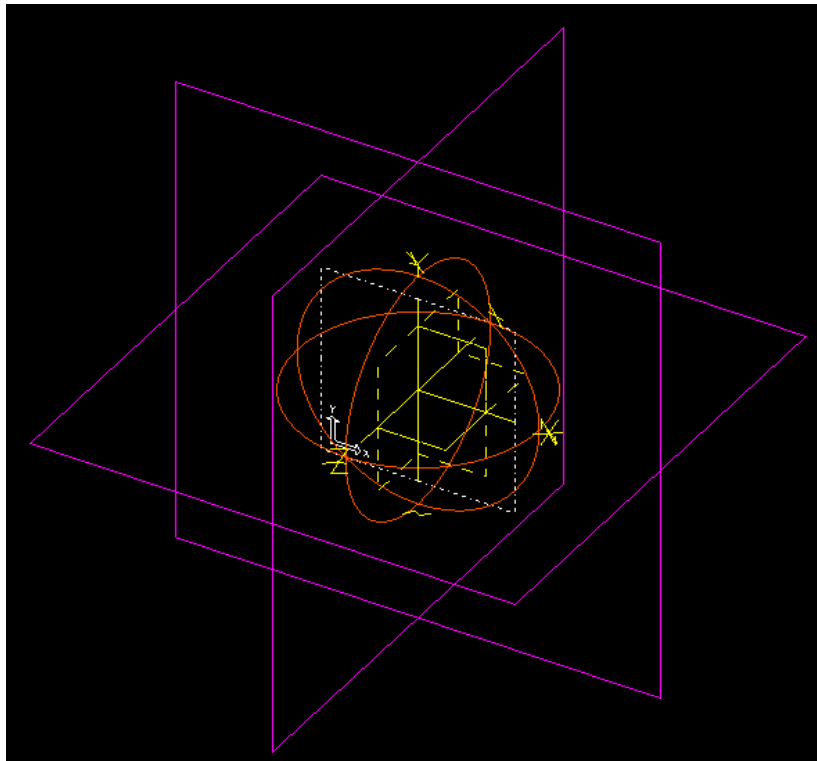
Click and hold on the parts icon  (manage bin) and  choose name part icon. Click on the block and name it as cube and name the sphere as ball. It is very important to make sure that cube and sphere should be named as mentioned else it would cause a problem in later stages while using the macro.

Click and hold on the parts icon again and choose  . Click on cube and press enter. This removes the cube from the screen but it will be available for operations and can be retrieved by clicking on manage bin.




IDEAS cannot mesh a sphere. To avoid overcome this problem the sphere has to be partition. For this a plane is created by clicking on  and choosing plane. The dimensions of the plane can be set as 0.5,0.5. The sphere has to partitioned in all three directions to enable meshing it.

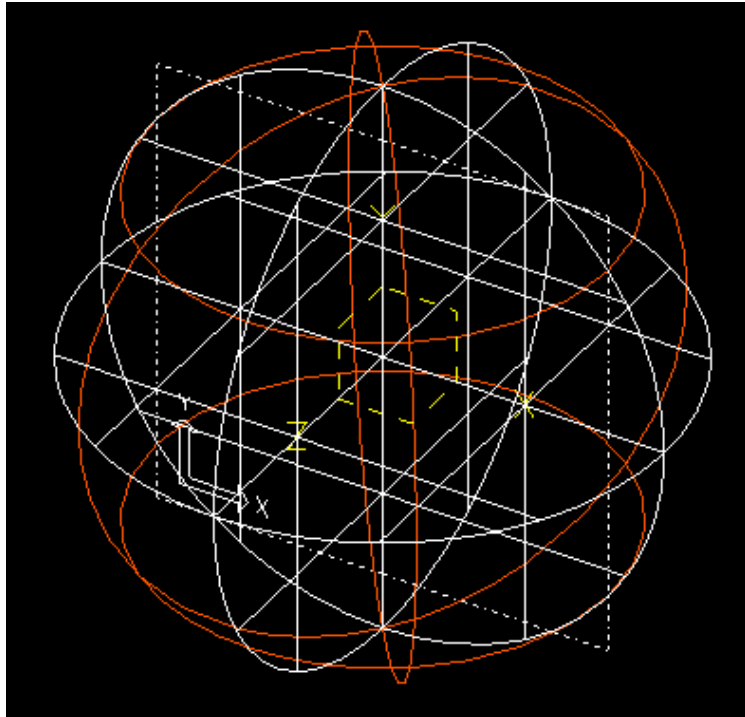
So we need 3 planes. These 3 planes are created by rotating the plane that we previously created. Click and hold on  and click on  that opens. Click on the plane that is created and hit enter. Choose origin as the center of rotation. Use the right click of the mouse on the main screen and select “copy sw” from the pop up menu and choose “on”.



Choose the direction of the rotation by clicking on “x axis” on the pop up menu and enter the angle of rotation as 90° and press enter. Repeat the same process about y axis. This is the







picture after the rotations.

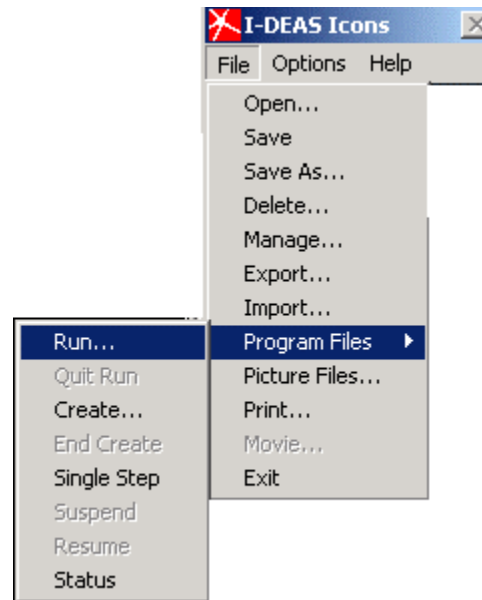
Click and hold  (cut) icon and click on  (partition) icon on the new menu. Choose one of the planes (any of them) as the partitioning part and the sphere as the part to be partitioned then press enter. The plane that was selected disappears and leaves the partitioned sphere behind. Repeat the same procedure with the remaining two planes. Partitioning introduces new surfaces in the sphere this will increase the number of elements in the mesh. In order to overcome this problem the extra surfaces introduced during partitioning are to be deleted. Click on  (delete) icon and right click in the main window click on “filter” on the pop up menu. This opens a new menu “selection filter”. Click on surface and click “ok”. Hold the shift key and carefully select the surfaces that are caused due to partitioning. There would be 12 surfaces in all that have to be deleted. After all surfaces are selected hit enter to delete the selected surfaces.



Now the click and hold on  icon choose  the icon click on cube and click on ok. This will bring the cube back onto the main screen.

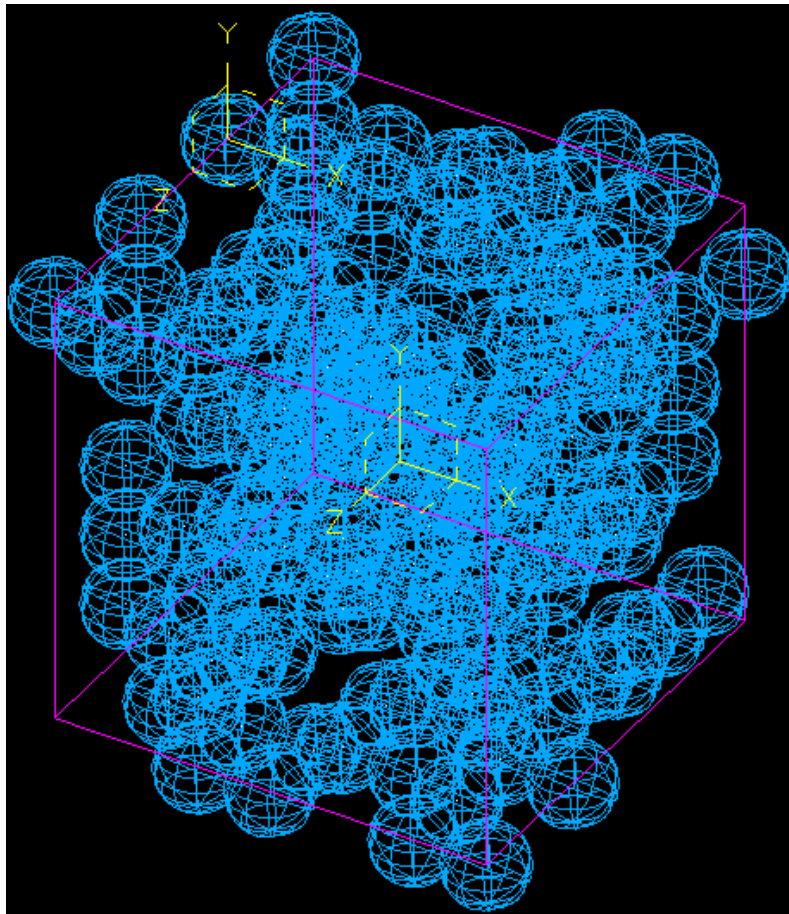
Now click and hold the  icon and choose  and click on the ball. This will remove the ball from the main screen and will put it in the bin. Click and hold the icon  and choose  icon. Click on the cube hit enter and choose “move to” on the pop up menu. Enter 1.728, 1.728, 1.728. This moves the center of the cube to numbers entered thus effectively moving the whole cube. Now get the ball to main screen repeating the get process. Now both ball and the cube are ready for the macro.


The macro is generated by running a FORTRAN program. Click on the file button and choose “program file” the click on “run” this opens a new window where the program file (the macro) is selected. Choose temp.prg. Click on “open” and then click on “ok”. The marco will start running generating a copies of the part ball and will move them to the locations specified in the macro. After generating the all the copies it will join them form a single part.





Note: For every run make sure that IDEAS file has no copies of ball from the previous run.

All the balls are joined as a single part. Now repeat the put away procedure & put the part named ball.



Now its time to remove the voids from the cube. For This click on the  (cut) icon.

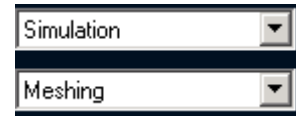
Pick all the joined spheres as cutter and cube is the part that is to be cut. This leaves behind the cube with voids. Click on  (measure) icon and .



Choose the cube with the voids cut and on the new menu click on calculate. The value in the volume column gives the volume of the solid present. To find the volume fraction of the voids removed subtract the value of volume remaining from the original volume to

get volume of the voids removed. To find the percentage of void space, divide the volume of voids removed by the total volume of the cube. Click on “file” and “save the file” with the volume fraction percentage in appropriate directory.

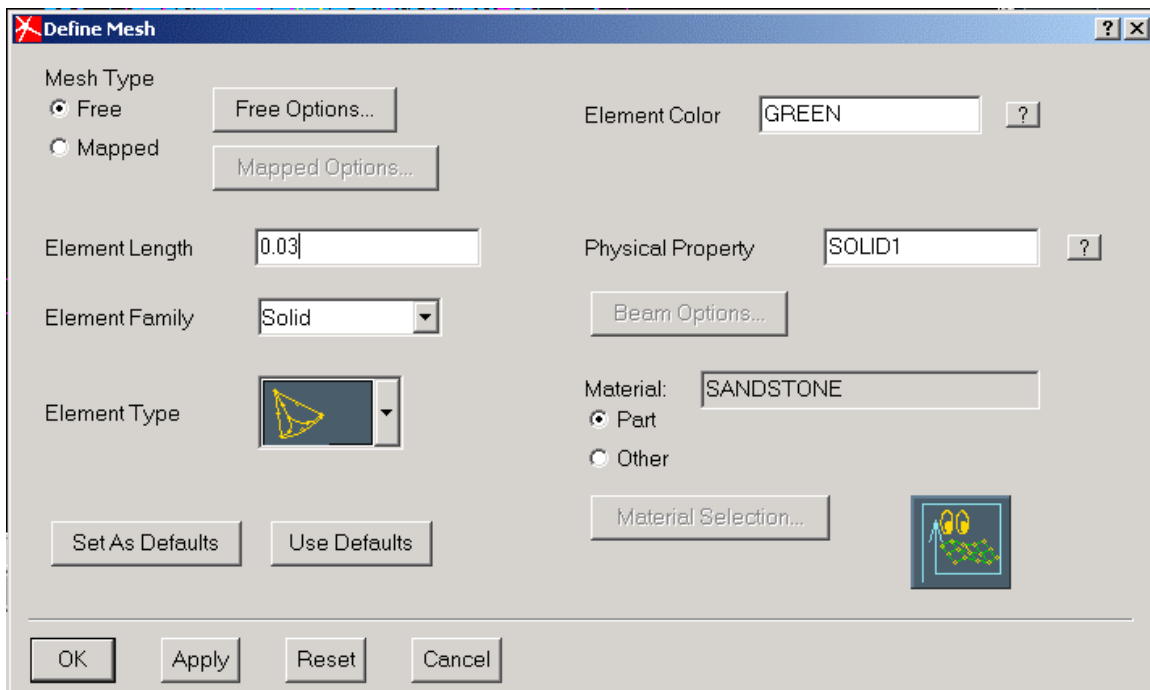
II.2 Meshing the Model


Choose “meshing”.









Click and hold  (define meshing) icon and click  on icon. Choose the cube as the part to be meshed. A new menu pops up showing the name of the part that's being meshed and the material that is being used for the mesh. Name and properties of the material are not set at this moment. They will be modified later on.

Click on ok.




Options are set as shown in the figure. Click on  . This starts preview meshing.

Sometimes the meshing is not completed due to the complex shape of the voids. So the element length has to be tweaked around till the meshing task is successful. After this is

done the meshing has to be checked for distortions. Click and hold the  (quality check) icon. Check distortions box on the pop up menu and set the value in between 0.3 to 0.7 and click on ok. The distorted elements are stored as a group. If there are any distorted elements click and hold  and choose  (move mid nodes). Right click on the main screen and click on “all done”. This removed most of the distortions. Then click on  (element) and choose  then right click and click on “all done” and hit enter. Now click on “order” and on the new pop up click on “straighten-----” and hit enter. Click on  and right click on the main screen and click on “filter” on the new pop up menu. Choose “nodes” on the menu and click on ok. Then right click on the main screen again and click on “all done” and hit enter twice. This deletes the unassociated nodes. Now repeat the element modify process this time only difference is that instead of choosing “straighten-----” we choose “parabolic -----” and hit enter. This removed all the distortions.

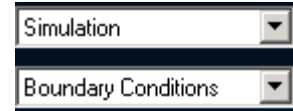
Note: Removal distortions is an important aspect because the model will not be solved I there are any distortions in the model.


Now comes setting the material properties. Click on  (materials) and the material menu opens up. Click on the material that is present and rename it as “Sandstone”. Then click on “modify”. This opens up the modify menu. Click on property and the properties menu opens up. Change the required material properties and click ok on all the menus.



Now the required mesh and material properties are rendered to the cube. Next stage is loading and restraining the cube.

II.3 Boundary Conditions

Choose “Boundary Conditions”.





Click and hold on  and click on display filter icon. Click in “fem entities” & uncheck boxes corresponding to “nodes”, “elements”, “displacement restrains” and “coupled degree of freedom”. This hides all the nodes, elements, displacement restrains and coupled degree of freedom so that the required entities can be selected.








Restraints are to be applied to make sure that there is no rigid body motion. Restraints are applied on 3 sides. Click and hold  on and choose  Displacement Restraint... . The displacement restrains are applied on nodes on the 3 surfaces. To select only the nodes on the surface right click on the main window and click on “filter” and select nodes on the new menu that opens. Then click on “related to”. This opens a new pop up menu choose “surface”. Now apply displacement restrains on the YZ surface that’s behind the Isometric view (which is the 3-D view of course). Now hold the F3 key and make sure the cursor is within the 80% region. Move the mouse until you can see the back YZ surface. Now click on the surfaces holding the shift key. Make sure all the surfaces are selected and then hit enter. Displacement restraint menu opens up. Rename the “restrain set 1” accordingly. Check “specified” and click on “specify restraint”. This opens the specify restraint menu. Click on “set all free”. Click on the drop box corresponding to X and set it as fixed and make sure the value is set to “zero”. This ensures that there is no displacement in the X direction.

Repeat this procedure for the XZ and the XY surface.

When new displacement restrains are applied IDEAS over writes the existing ones. So when displacement restrains are applied on XZ surface the displacement nodes on Z axis bottom edge are replaced by displacement restrain along Y axis. But actually on the Z axis bottom edge we need restrains to be in both X and Z directions.

So we have to repeat the applying displacement restrain process with the exception that now instead of choosing the surface we choose edge and choose the Z axis. While specifying the restrains repeat the same process the only difference being now the restrains on both X and Z directions are set to zero. This process of redefining the restrains has to be performed on the other 2 edges too. After this is done there is one more manipulation to perform. The back vertex has to be restrained in all three directions. While picking the edge perform the same operations as performed earlier. The only difference is this time instead of clicking on surface click on “edge”. So repeat the restraining process and this time the displacement of the vertex is set to “zero” by setting all the 3 drop boxed to fixed. Now all the nodes are restrained accordingly.

Now force is applied on the other 3 faces. The nodes need to be coupled so that the effect of the force applied on a single node is similar to applying pressure on the face. Click and hold the FEM groups  icon and click on the same icon this opens a new set of icons called FEM Groups. Now all the nodes belonging to one surface are to be added into a group. Click on  (add to group). To select only the nodes on the surface right click on the main window and click on “filter” and select nodes on the new pop up menu that opens. Then click on “related to”. This opens a new menu choose “surface”. Now click on the front YZ surface make sure that all the surfaces are selected and hit

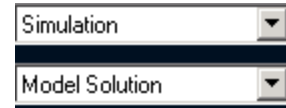
enter. Name the group as “YZ face”. Repeat this process for the other 2 faces also forming groups called “ZX face” and “XY face”. “XY face” is the group that was created last so the current group is “XY face”. Click on “display filter” and click on “FE entities” and check the “nodes”. Now only the nodes that are stored in the “XY face” are displayed. Click on add to group again and choose any node preferably near the center and hit enter. Name the new group as “Z force”. Only the single node is on display on the main window. Click on (set current group)  and click on “XY face”. Now click on  (display current group). This brings the nodes of “XY face” to display. Now we need to remove the node “Z force” from “XY face”. Click on  (remove from group) and right click on the main window and choose “Z force”. Click and hold on  (displacement restrain) and click on  Coupled DOF... to apply coupled degree of freedom on “XY face” and “Z force”. First the independent node is to be selected. Right click on the main window and choose click on “groups” and then click on “Z force” on the new pop up menu. To set the dependent nodes right click on the main window and choose click on “groups” and then click on “XY face” on the new pop up menu. Now we need to apply load on “Z force”. Click on  (load). Right click on main window and click on “groups” and choose “Z force” on the new pop up window. “Force on node” menu opens up. Click on “load set 1” and rename it accordingly. Click on “Z force” and set its value to -15668. The negative sign signifies that the force is compressive in nature. Repeat this process for applying coupled degree of freedom for the other 2 groups i.e. “YZ face” and “ZX face”. Click on  (boundary condition set). “Boundary condition set management” pop up menu opens. **Do not** change anything in the drop box having “linear static”. Check the restrain set & constrain set check boxes and makes sure



the sets are the ones named earlier. Click on load set too and click on “apply” and the click on “ok”.

Note: It's very important to save the changes as soon as possible because if there the computer shuts down for some reason all the work will be lost. But be cautious if u want to make any modifications since no more modifications are possible after the save.

II.4 Solution

Choose “Solution”.



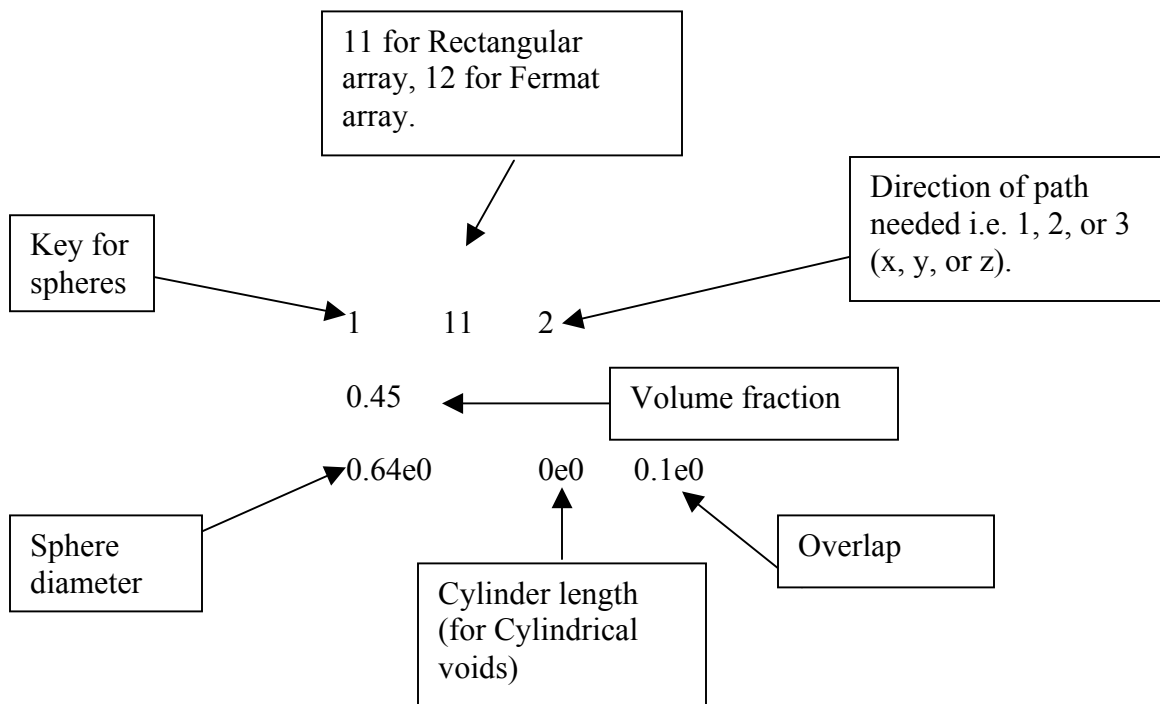
Click on  (solution set). On the “solution control” check to make sure that the name of the boundary condition set is same as the one named earlier also make sure that the type of solution is “liner static”. Click on “ok”. Click on and hold  (start solution icon) and choose the same icon on the menu that pops up. A new menu will open, click on “ok”. Now the model file is being solved.

Appendix III

III.1 Creating the Model

Open Fortran90 and run the compile the file “randomcylinder.for”. In the Beginning of the program look for the path of the input file. Look for the following line, “OPEN (11,file='c:\matlab\bin\pream\input1.txt')”, or set the path of the input file in the program accordingly.

Write the following input file in notepad, and save it as input1.txt.



Note: Every time any changes made in the `input1.txt` have to be saved to ensure that the changes that have been made are taken into consideration while running the program.

Run the Fortran file and look in the program for the path of the *.JOU file. Look for “open (62,file='c:\fluent-test\pream2\pream1.jou.txt')”, or set the path accordingly in the FORTRAN program. Temp.jou.txt has all the gambit commands to generate the whole model.

Now open a command prompt window and type “gambit”. Gambit application will open.

Click on **file** and click on **run journal**.



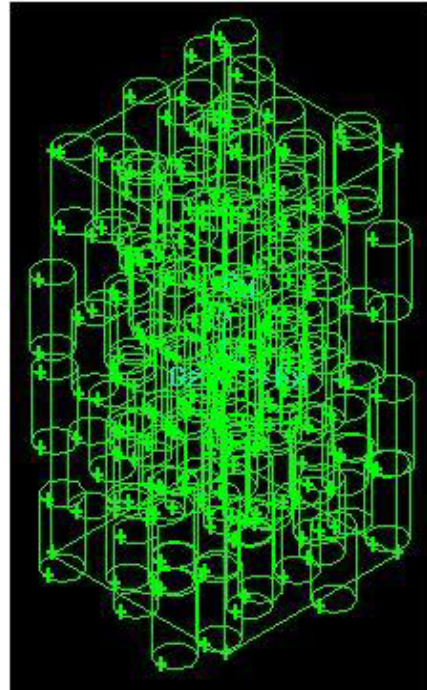
This will open a new window.


Click on browse and choose the .jou.txt file generated by the FORTRAN program, and click accept. Gambit will generate

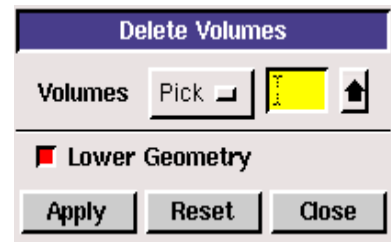
the model according to temp.jou.txt file.




The model would look like this after executing the journal file.



Now click on  (delete). On the menu that opens, click on the arrow and choose cylinder and click on the → arrow and click close and then on apply. Now the volume called “cylinder” is deleted.

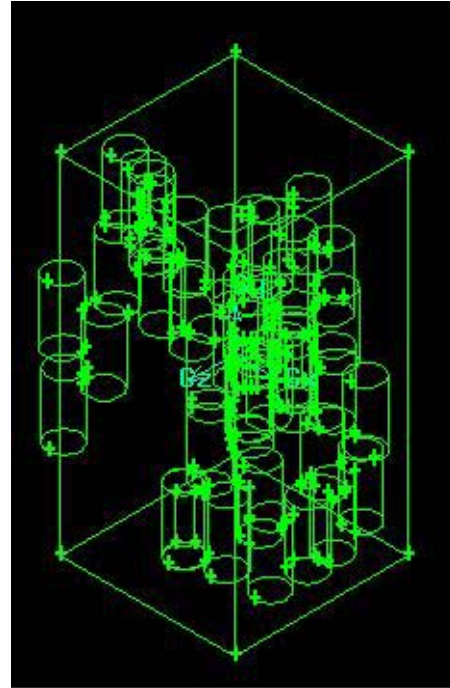



Click on  (unite). Click on All→. Choose the volume called “cube”, and click on ←. Click on close and then click on apply. Now all the voids that are connected are joined.



Note: If two voids are not touching each other they will remain as two separate volumes.

Delete the unnecessary volumes (the ones that do not make a path from the top to bottom). This has to be performed manually by looking at each volume and making sure the volume in consideration results in a path.

Click on the delete icon again and click on each volume and check the graphic window to see if it results in a path. If a volume doesn't form a path delete it. After the unnecessary volumes are deleted, the volumes left would look like the picture shown.





Click on the  (split volumes) icon. In the menu that pops up choose cube for volume 1 and chose the other volumes as the volume to split with. All volumes protruding out of the cylinder are removed forming inlets and outlets.

The direction of flow is along the Y-axis. So name the top surfaces (the cylinders that protruded out of the cube act as inlets and outlets) that are inlets as inlet1, inlet2 and so on. Repeat the process for bottom surface and name them as outlet1, outlet2 etc. Click on  (surface) icon. Now the labels are to be renamed. Click on  (modify label) icon. Click on each face or choose from the list and make sure only the surface is

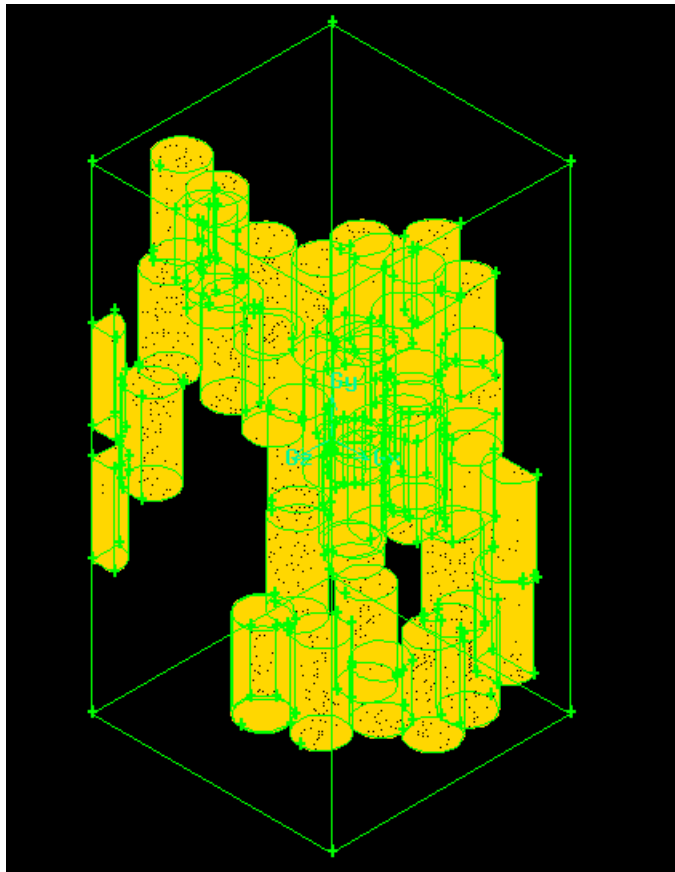
selected. When a surface that is to be set as inlet is highlighted label it as inlet1 etc. If the cylinders are protruding on sides name them as sides.

III.2 Meshing



Now the model is ready to be meshed. Click on  (mesh) icon. Click on the  icon. In the new menu that pops up choose the volume that forms the path and choose element as Tet/Hybrid, type is Tgrid and interval=0.05.

This how the RVE is going to look like after it is meshed.

Go to Solver and select FLUENT 5/6.

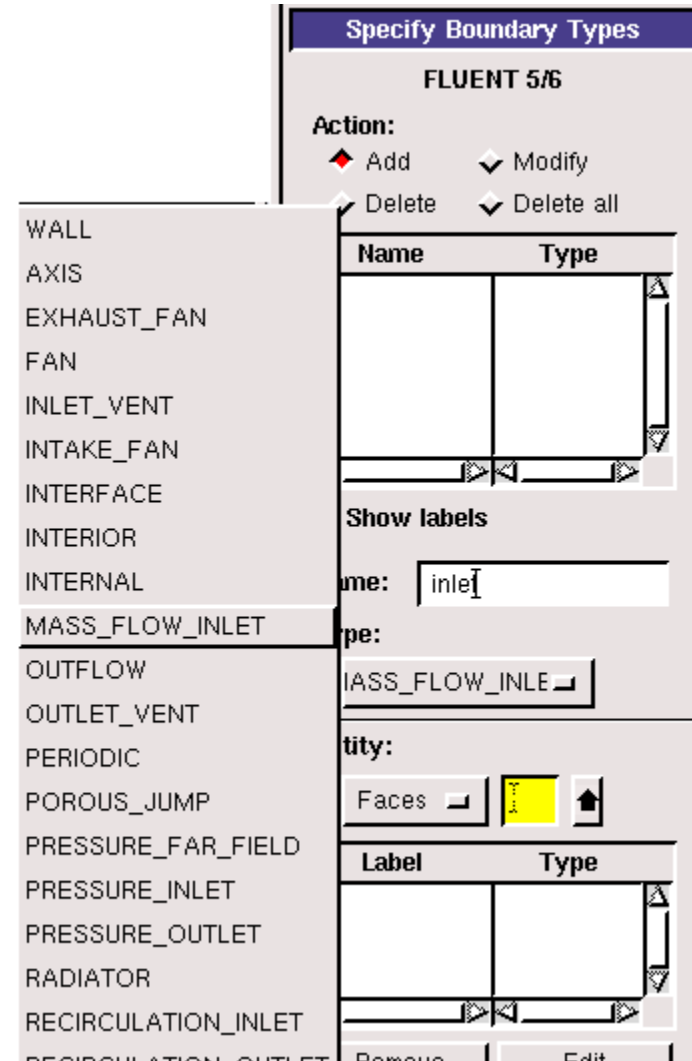



III.3 Boundary conditions

Click on  (zones) icon. Click on  (Boundary conditions) icon. Now the specify boundary types menu pops up.

In the name column fill in inlet.

Right click and hold on the type drop box and set it to mass_flow_inlet. Click on the arrow below entity and choose all the inlet faces (that were renamed previously). Click on apply. Fill in outlet in the name column and the type is set to pressure_outlet. The outlet faces are selected. Click on ok. Then repeat the process for sides and set symmetry boundary conditions for them.



Click on  (specify continuum types) icon. Set the name to fluid. Set the type to fluid and choose the volume that forms the path.

Note: If no continuum is set to the remaining volume Fluent is going to treat this volume as non-conductive.

Click on file→export→mesh and save the file as new.msh.

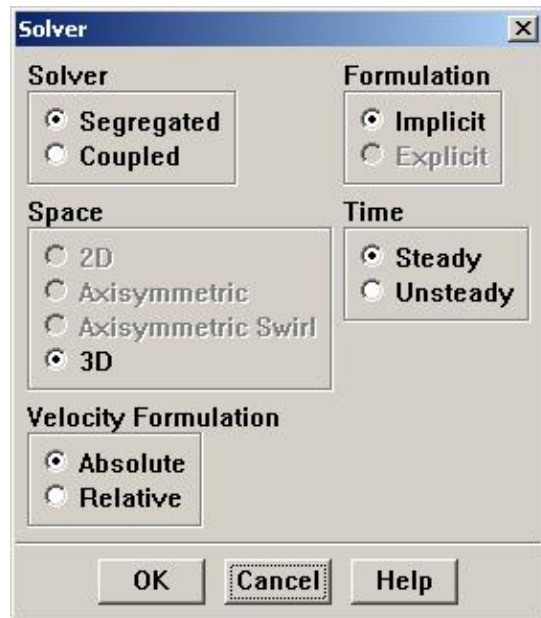
III.4 Fluent Model set up

Go to command prompt and type fluent and hit enter. Fluent application will start.

Click on file→ read→ case → open the new.msh file. To ensure the best possible grid quality for the calculation, it is good practice to smooth a triangular or tetrahedral grid after you read it into FLUENT. Click on the Smooth button and then click on Swap repeatedly until FLUENT reports that zero faces were swapped. Click on close. If FLUENT cannot improve the grid by swapping, no faces will be swapped.

Now click on Grid→ scale. Enter the 0.001 in the columns X, Y and Z of scale factor and click on scale.

Click on Define→ models→ solver. Check and make sure all options are set as shown in figure.



Click on Define →models→ viscous and make sure laminar is checked.

III.5 Defining material in Fluent.

Click on Define → Materials.

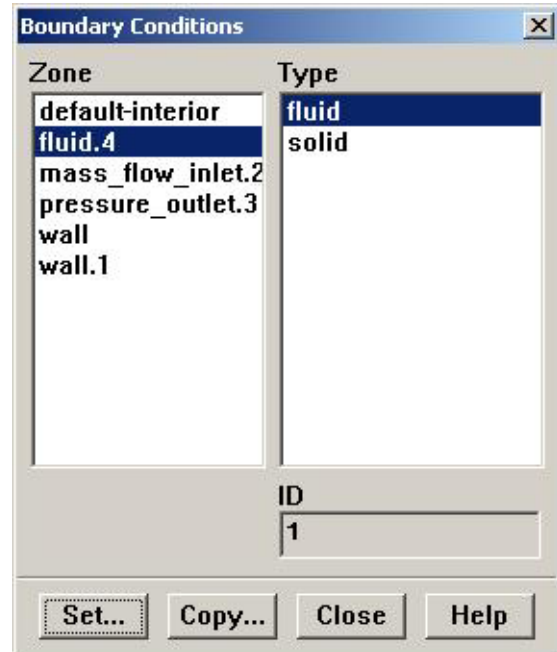
The screenshot shows the 'Materials' dialog box in ANSYS Fluent. The 'Name' field contains 'air'. The 'Material Type' dropdown is set to 'fluid'. The 'Order Materials By' section has 'Name' selected. The 'Chemical Formula' field is empty. The 'Fluid Materials' dropdown is set to 'air'. The 'Database...' button is visible. The 'Properties' section shows 'Density (kg/m3)' set to 'constant' with a value of '1.229' and 'Viscosity (kg/m-s)' set to 'constant' with a value of '1.7894e-05'. Both properties have 'Edit...' buttons. At the bottom are 'Change/Create', 'Delete', 'Close', and 'Help' buttons.

Field	Value
Name	air
Material Type	fluid
Order Materials By	Name
Chemical Formula	
Fluid Materials	air
Density (kg/m3)	constant 1.229
Viscosity (kg/m-s)	constant 1.7894e-05

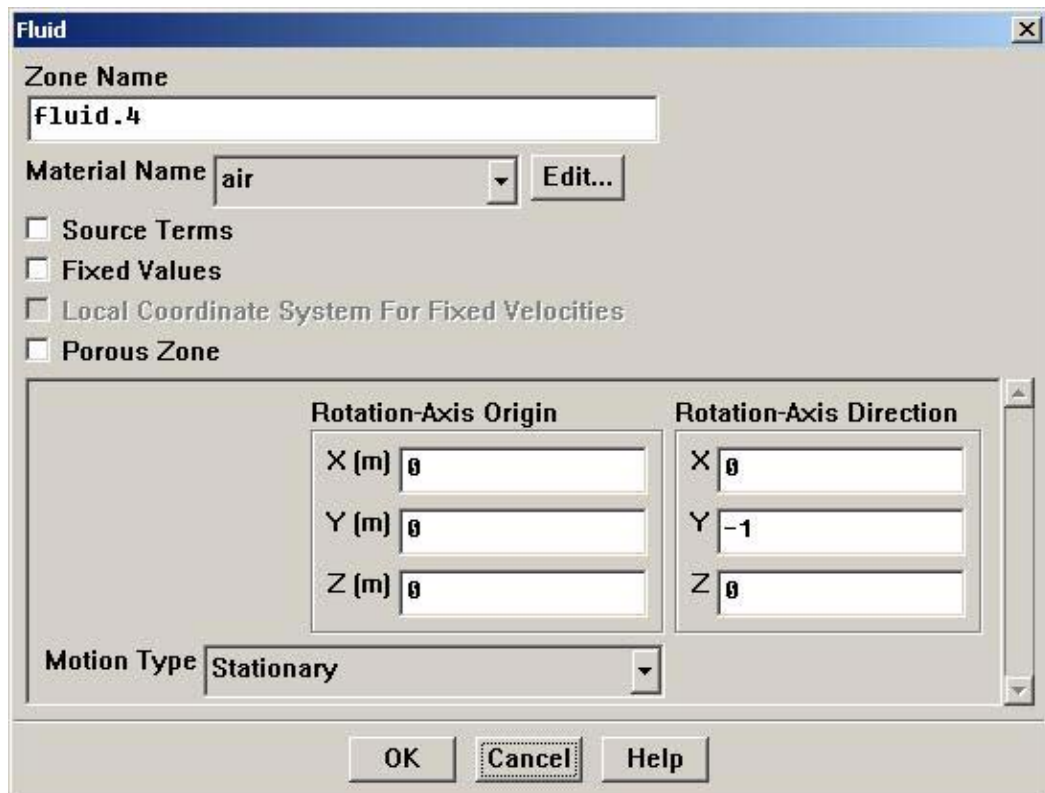
Click on fluid materials and choose air. The value of density is 1.229 kg/m^3 and the value of viscosity is set to $1.7894\text{e-}05$.

III.6 Boundary conditions

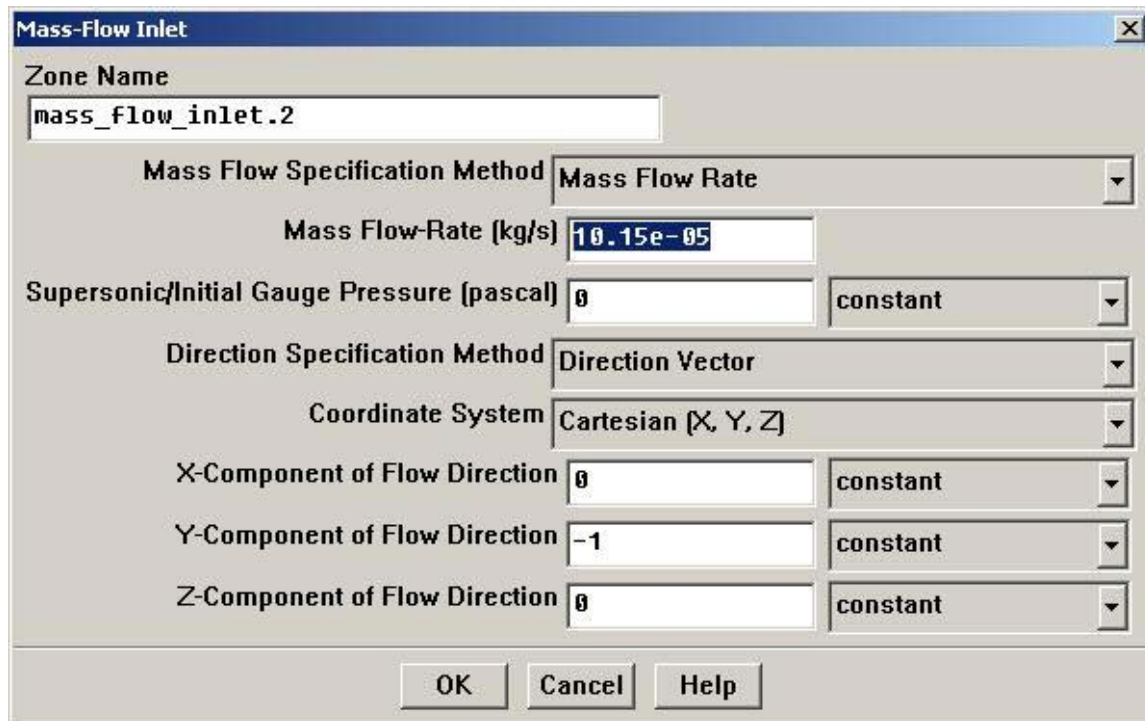
Click on define→ boundary conditions.
Click on fluid.4 under Zone and click on fluid under Type. Click on Set. Fluid menu pops up.



Set rotation axis direction value of Y to -1 and that of Z to 0.



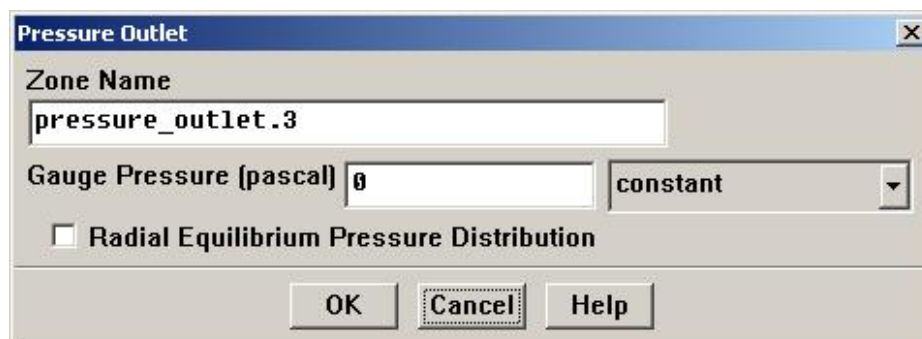
Now click on mass_flow_inlet.2 under zone (in Boundary conditions menu) and click on mass-flow-inlet under type. Click on set. Mass-flow Inlet menu opens. Set the value of



The 'Mass-Flow Inlet' dialog box is shown. It has a title bar with a close button. The 'Zone Name' field contains 'mass_flow_inlet.2'. The 'Mass Flow Specification Method' is set to 'Mass Flow Rate'. The 'Mass Flow-Rate (kg/s)' field contains '10.15e-05'. The 'Supersonic/Initial Gauge Pressure (pascal)' field contains '0' and the unit is set to 'constant'. The 'Direction Specification Method' is set to 'Direction Vector'. The 'Coordinate System' is set to 'Cartesian [X, Y, Z]'. The 'X-Component of Flow Direction' is '0', 'Y-Component of Flow Direction' is '-1', and 'Z-Component of Flow Direction' is '0', all with units set to 'constant'. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

mass flow rate to 10.15e-05. Set Y-component of flow direction to -1 and the rest to 0.

Click on pressure_outlet.3 under zones and click on pressure-outlet under types. Click on set. Pressure outlet menu opens.

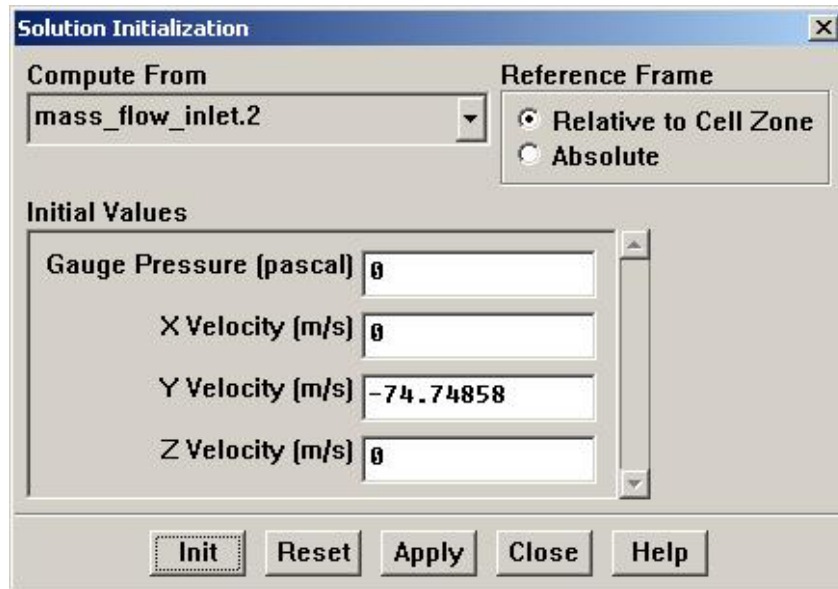


The 'Pressure Outlet' dialog box is shown. It has a title bar with a close button. The 'Zone Name' field contains 'pressure_outlet.3'. The 'Gauge Pressure (pascal)' field contains '0' and the unit is set to 'constant'. There is a checkbox for 'Radial Equilibrium Pressure Distribution' which is currently unchecked. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

Click on ok. All the required boundary conditions are set.

III.7 Solution

Click on solve → Initialize → Initialize. The Solution initialization menu opens up.

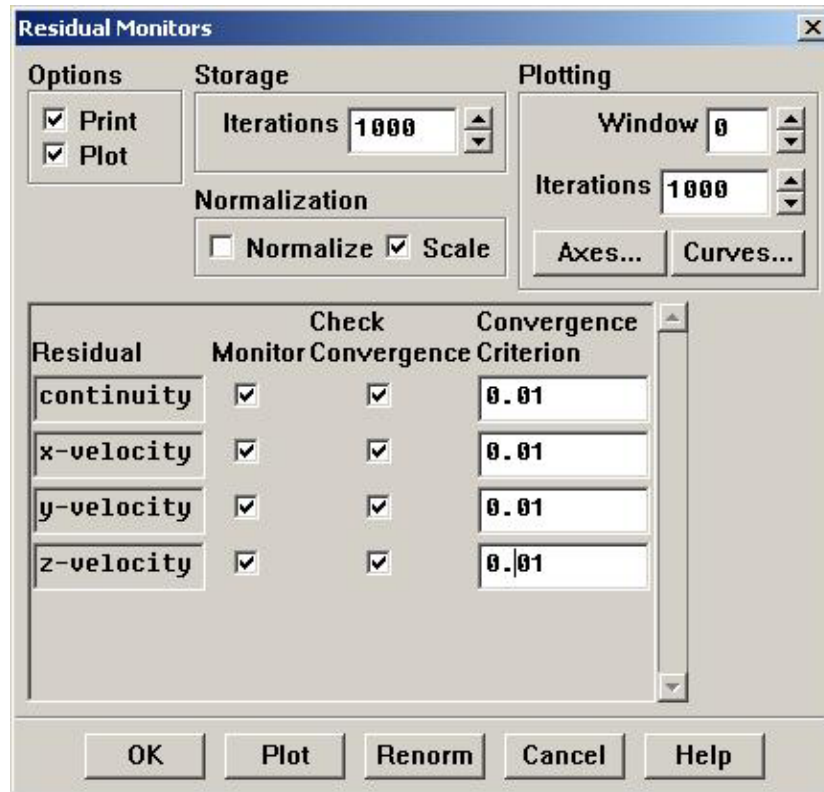


Click on the drop box corresponding to compute from, and choose mass_flow_inlet.2.

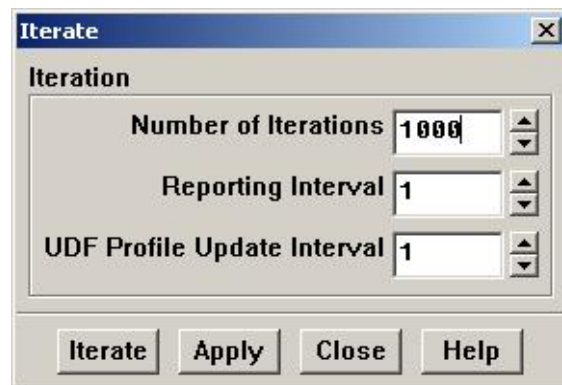
Click on Init and click on close.

To enable plotting of residuals during solution click on solve → monitors → residuals.

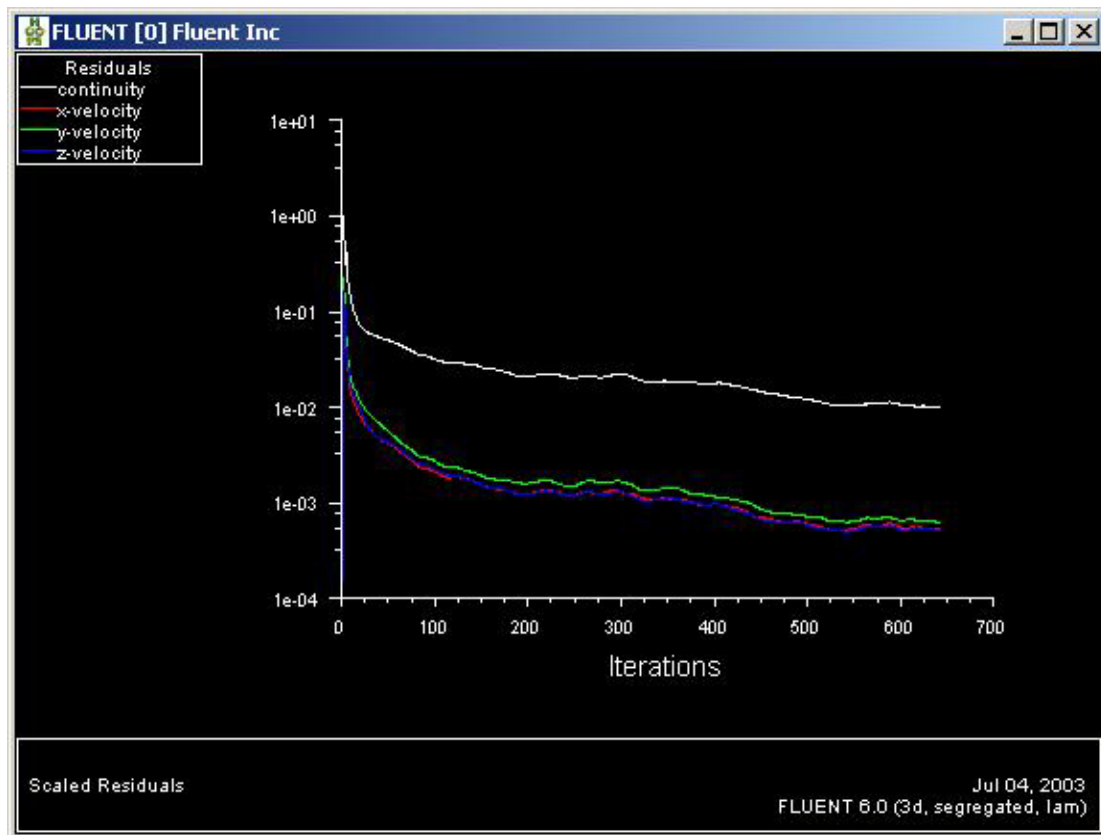
Check Plot and set the convergence criterion to 0.01 for continuity, X-velocity, Y-velocity and Z-velocity. Click on Ok.



Click on Solve→Iterate. Enter number of iterations to 1000. Click on Apply. Click on iterate to start the iterations.



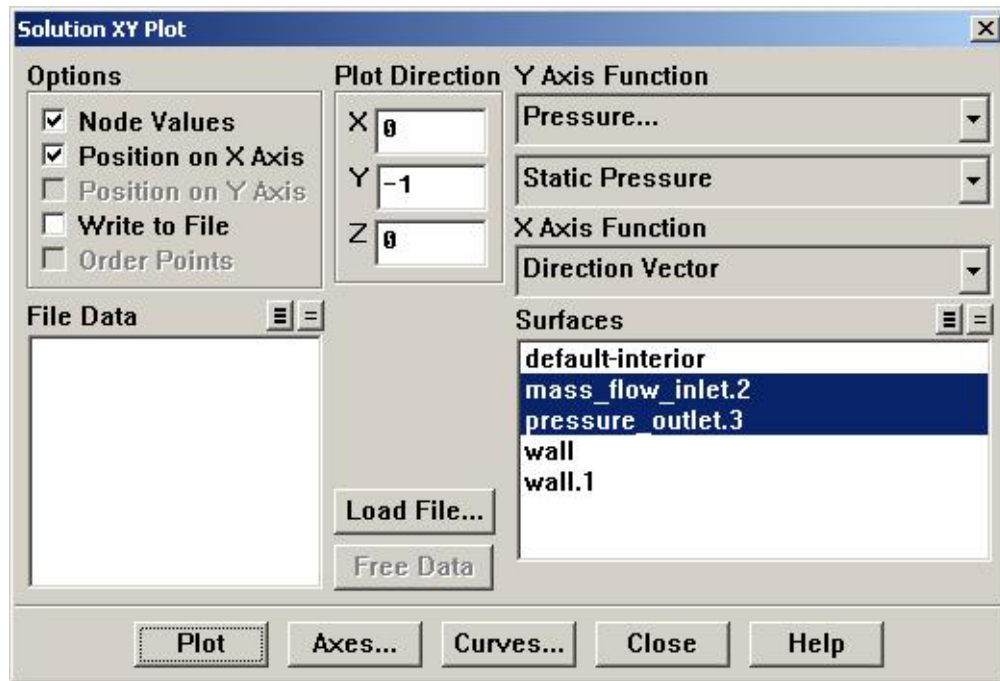
The solution after convergence will look similar to the figure shown below



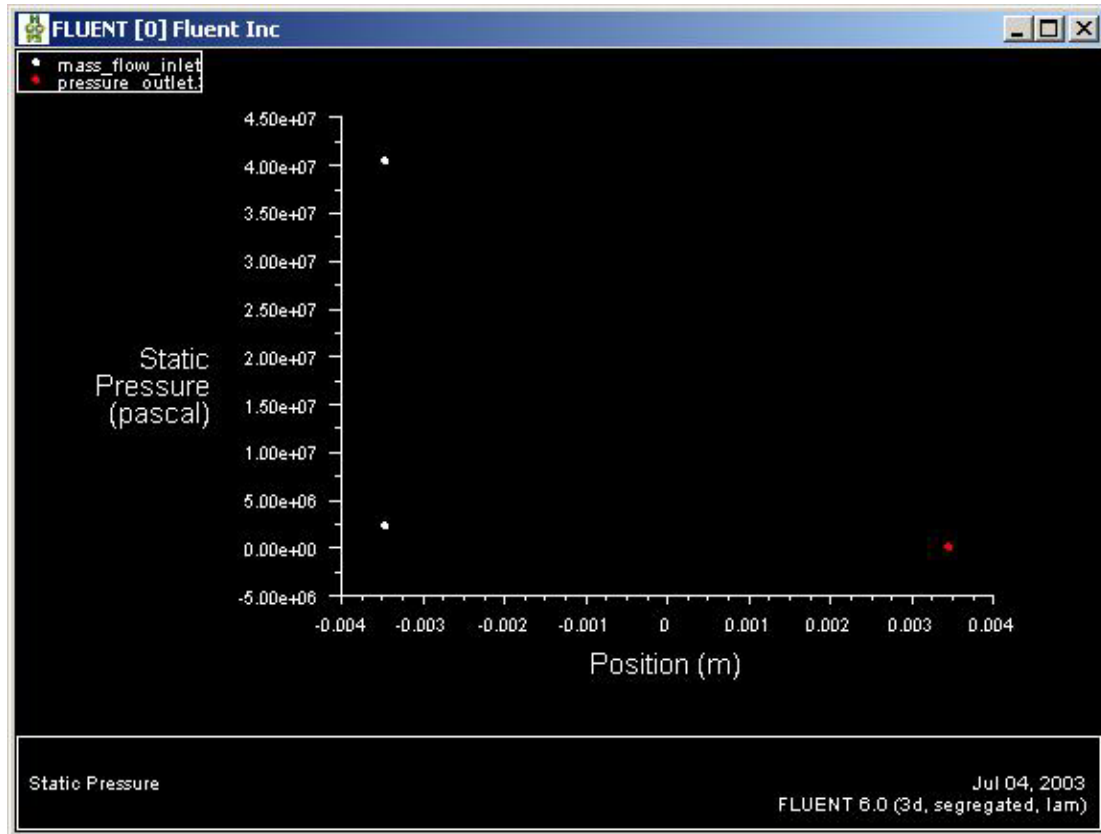
Then to save the results click on file→ write→ data

III.8 Displaying results

Click on Plot → XY plot.



Change plot direction of Y to -1 and rest are set to 0. Make sure the Y-axis function reads pressure. Click on mass_flow_inlet.2 and pressure_outlet.3. Click on plot.



The white dots indicate the pressure at the inlet and the red dot is the outlet pressure.

Take an average of the white dots to obtain the inlet pressure. It is named as P_i . The value of red dot is named as P_o .

$$\Delta P = (P_i + P_o) / 2$$

The permeability is evaluated using the MathCAD program (Appendix IV).

Appendix IV

For 40 % Cylindrical Voids Fermat Array

$$\begin{aligned} \rho &:= 850 & \mu &:= 5 \cdot 10^{-3} & \text{rad} &:= 0.32 \cdot 10^{-3} \\ a_1 &:= 3.45 \cdot 10^{-3} & a_2 &:= 6.912 \cdot 10^{-3} & a_3 &:= 3.45 \cdot 10^{-3} \\ L &:= a_2 & A_c &:= a_1 \cdot a_3 & \text{Mass_flow} &:= 10.15 \cdot 10^{-5} \end{aligned}$$

$$\Delta P_1 := 7.41 \cdot 10^3 \quad q_1 := \frac{\text{Mass_flow}}{\rho} \quad q_1 = 1.194 \times 10^{-7}$$

■ + ■

$$K_1 := \mu \cdot L \cdot \frac{q_1}{A_c \cdot \Delta P_1} \quad \text{m}^2 \quad K_{\text{new1}} := K_1 \cdot 1.013 \cdot 10^{15} \quad K_{\text{new11}} := K_1 \cdot 10^{12}$$

$$K_{\text{new1}} = 4.74 \times 10^4 \quad \text{mDarcy} \quad K_{\text{new11}} = 46.791 \quad \mu\text{m}^2$$